Synchronized Illumination Modulation for Digital Video Compositing

von

Anselm Grundhöfer

Dissertation eingereicht an der

Fakultät Medien

zum Erreichen des akademischen Grades

Dr.-Ingenieur (Dr.-Ing.)

an der

BAUHAUS-UNIVERSITÄT WEIMAR

Gutachter	Univ. Prof. Dr.-Ing. habil. Oliver Bimber Johannes Kepler Universität Linz, Österreich
	Prof. Mark Billinghurst HITLab, Christchurch, Neuseeland Zweitgutachter
Tag der Einreichung	26. August 2010

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Weitere Personen waren an der inhaltlich-materiellen Erstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich hierfür nicht die entgeltliche Hilfe von Vermittlung- bzw. Beratungsdiensten (Promotionsberater oder anderer Personen) in Anspruch genommen. Niemand hat von mir unmittelbar oder mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer anderen Prüfungsbehörde vorgelegt. Ich versichere, dass ich nach bestem Wissen die reine Wahrheit gesagt und nichts verschwiegen habe.

Anselm Grundhöfer, Weimar, den 26. August 2010

ABSTRACT

Besides home entertainment and business presentations, video projectors are powerful tools for modulating images spatially as well as temporally. The re-evolving need for stereoscopic displays increases the demand for low-latency projectors and recent advances in LED technology also offer high modulation frequencies. Combining such high-frequency illumination modules with synchronized, fast cameras, makes it possible to develop specialized high-speed illumination systems for visual effects production.

In this thesis we present different systems for using spatially as well as temporally modulated illumination in combination with a synchronized camera to simplify the requirements of standard digital video composition techniques for film and television productions and to offer new possibilities for visual effects generation.
After an overview of the basic terminology and a summary of related methods, we discuss and give examples of how modulated light can be applied to a scene recording context to enable a variety of effects which cannot be realized using standard methods, such as virtual studio technology or chroma keying. We propose using high-frequency, synchronized illumination which, in addition to providing illumination, is modulated in terms of intensity and wavelength to encode technical information for visual effects generation. This is carried out in such a way that the technical components do not influence the final composite and are also not visible to observers on the film set. Using this approach we present a real-time flash keying system for the generation of perspectively correct augmented composites by projecting imperceptible markers for optical camera tracking. Furthermore, we present a system which enables the generation of various digital video compositing effects outside of completely controlled studio environments, such as virtual studios. A third temporal keying system is presented that aims to overcome the constraints of traditional chroma keying in terms of color spill and color dependency.

In addition to the development and evaluation of various applications for digital video compositing, such as alpha matting, chroma keying optimizations, environment matting, imperceptible optical camera tracking and in-scene moderation support, user studies were also carried out to analyze whether the high-frequency temporal image modulations affect observers.

DEUTSCHE ZUSAMMENFASSUNG

Einleitung

Informationsaustausch ist eines der Grundbedürfnisse der Menschen. Während früher dazu Wandmalereien, Handschrift, Buchdruck und Malerei eingesetzt wurden, begann man später, Bildfolgen zu erstellen, die als sogenanntes ”Daumenkino” den Eindruck einer Animation vermitteln. Diese wurden schnell durch den Einsatz rotierender Bildscheiben, auf denen mit Hilfe von Schlitzblenden, Spiegeln oder Optiken eine Animation sichtbar wurde, automatisiert – mit sogenannten Phenakistiskopen, Zoetropen oder Praxinoskopen.

Mit der Erfindung der Fotografie begannen in der zweiten Hälfte des 19. Jahrhunderts die ersten Wissenschaftler wie Eadweard Muybridge, Etienne-Jules Marey und Ottomar Anschütz, Serienbildaufnahmen zu erstellen und diese dann in schneller Abfolge, als *Film*, abzuspielen. Mit dem Beginn der Filmproduktion wurden auch die ersten Versuche unternommen, mit Hilfe dieser neuen Technik spezielle visuelle Effekte zu generieren, um damit die Immersion der Bewegtbildproduktionen weiter zu erhöhen. Während diese Effekte in der analogen Phase der Filmproduktion bis in die achtziger Jahre des 20.Jahrhunderts recht beschränkt und sehr aufwendig mit einem enormen manuellen Arbeitsaufwand erzeugt werden mussten, gewannen sie mit der sich rapide beschleunigenden Entwicklung der Halbleitertechnologie und der daraus resultierenden vereinfachten digitalen Bearbeitung immer mehr an Bedeutung. Die enormen Möglichkeiten, die mit der verlustlosen Nachbearbeitung in Kombination mit fotorealistischen, dreidimensionalen Renderings entstanden, führten dazu, dass nahezu alle heute produzierten Filme eine Vielfalt an digitalen Videokompositionseffekten beinhalten. Vergleicht man den Erfolg aktueller Kinoproduktionen, stellt man fest, dass ein Großteil der Erfolgreichsten auch die Academy Awards für ihre visuellen Spezialeffekte erhielten [115, 144]. Dies verdeutlicht, dass die Popularität und damit auch die Lukrativität von Filmproduktionen, neben weiteren Faktoren, auch mit dem Einsatz von professionellen Spezialeffekten korreliert.

Obwohl visuelle Effekte nun schon seit Jahrzehnten in der digitalen Filmproduktion eingesetzt werden, bedürfen sie immer noch eines enormen manuellen Bearbeitungsaufwandes und sind aufgrund der eingesetzten Techniken in ihrer Flexibilität beschränkt. Jeder einzelne professionell erstellte visuelle Effekt bedient sich zwar meist an Standardtechniken wie zum Beispiel dem Chromakey-Verfahren zur Berechnung der notwendigen Bildinformationen, benötigt jedoch immer noch einen nicht zu unterschätzenden Anteil an manueller Nachbearbeitung, um das finale Composite realis-

tisch erscheinen zu lassen.

Während das Chromakey-Verfahren zum de-fakto Standard für die Aufteilung des Bildes in Vordergrund- und Hintergrundbereiche geworden ist, zeigt das Verfahren Einschränkungen bezüglich Farbabhängigkeit und Colorspills. Da die Vordergrund-/Hintergrundseparation aufgrund der vorherrschenden Farbtöne berechnet wird, ist es offensichtlich, dass Vordergrundobjekte eine dem Hintergrund unähnliche Färbung benötigen, da sie sonst fälschlicherweise dem Hintergrund zugeordnet werden würden. Mit Colorspill wird das ungewollte Vorhandensein der zum Keying verwendeten Hintergrundfarbe im finalen Composite bezeichnet. Dies kann einerseits in halbtransparenten Regionen, wie Haar, auftreten, aber auch durch indirekte Reflexionen der Hintergrundfarbe auf Vordergrundobjekten.

Die größtenteils automatische Erstellung von visuellen Effekten außerhalb von vollkommen kontrollierten Umgebungen wie virtuellen Studios stellt auch heutzutage noch eines der grundlegenden Probleme der digitalen Videokomposition dar und kann meist nur mit einem nicht unerheblichen manuellen Aufwand durchgeführt werden: Während der Aufnahme werden teilweise Hintergrundbereiche mit mobilen Blue- oder Greenscreens abgedeckt und die Szene mit automatisierten Kamerafahrten mehrmals aufgenommen, was die Kreativität solcher Szenen einschränkt. Während der Postproduktion müssen die fehlenden Teile des Szenenhintergrundes vollkommen durch andere Bildinhalte oder das mehrmals aufgenommene Videomaterial nahtlos ersetzt werden, was den gesamten Produktionsaufwand beachtlich erhöht.

Das beschränkte zeitliche Auflösungsvermögen der menschlichen visuellen Wahrnehmung wird im Kontext von Filmproduktionen offensichtlich dazu genutzt, aus einer Reihe von Standbildern den Eindruck von kontinuierlichen Bewegungen zu erzeugen. Andere Techniken, wie Zoetrope oder das projektionsbasierte Morphovision Display [59] machen sich das beschränkte zeitliche Auflösungsvermögen des menschlichen Auges zu Nutzen, um statische Objekte durch stroboskopische Beleuchtung animiert erscheinen zu lassen. Analysiert man die aktuelle Betrachtungssituation, kann man diese Beschränkung auch dazu nutzen, in der Beleuchtung bestimmte Informationen vor dem menschlichen Auge zu verstecken, sie jedoch einer synchronisierten Kamera zur Verfügung zu stellen und somit zum Beispiel die Grundlagen für die Generierung von visuellen Effekten zu schaffen ohne dass ein Betrachter diese technischen Informationen wahrnimmt. Da die Sichtbarkeit dieser versteckten Informationen jedoch von einer Reihe von Faktoren beeinflusst wird, ist eine vollständige Unsichtbarkeit in alles Betrachtungssituationen bisher nicht möglich gewesen.

Die in diesem Zusammenhang angesprochenen Probleme und Einschränkungen der digitalen Videokomposition mit Hilfe von spezieller, zeitlich und räumlich modulierter Beleuchtung zu überwinden stellt die Basis dieser Dissertation dar. Im Folgenden werden eine Reihe von Möglichkeiten aufgezeigt wie durch den Einsatz solcher Systeme die Erstellung visueller Effekte vereinfacht und neue Möglichkeiten der digitalen Videokomposition geschaffen werden können.

Motivation

Mit der fortschreitenden Entwicklung der Projektionstechnologien werden immer schnellere Bildwiederholraten ermöglicht. Während der gerade wieder entstehende Trend zur stereoskopischen Filmwiedergabe die Nachfrage nach schnellen Projektoren mit niedrigen Latenzzeiten antreibt, erlaubt die moderne LED-Technologie eine noch wesentlich höhere Modulationsfrequenz. Die Kombination solcher hochfrequenten Beleuchtungselemente mit schnellen Kamerasensoren ermöglichte die Entwicklung von speziellen Hochgeschwindigkeitssystemen wie der Lightstage [44], mit der in einer vollkommen kontrollierten Umgebung eine Reihe von qualitativ hochwertigen visuellen Effekten, wie der bildbasierten, synthetischen Beleuchtungssimulation, realisiert werden können. In Kombination mit dem enormen Anstieg an paralleler Rechenleistung moderner Grafikkarten und den daraus resultierenden Möglichkeiten der Echtzeit-Bildverarbeitung können solche hochfrequenten Systeme zur Generierung von Live-Videokompositionen genutzt werden.

Heutzutage wird die virtuelle Studiotechnologie in vielen Film-und Fernseh-Produktionen zur Erstellung von visuellen Effekten eingesetzt. Die erforderliche Auftrennung des Videobildes in Vordergrund- und Hintergrundregionen wird mit dem Chromakey-Verfahren realisiert, welches einen homogen beleuchteten, üblicherweise blauen oder grünen Hintergrund erfordert. Das aufgezeichnete Video-Signal wird dann entweder online oder in der Nachbearbeitung mithilfe von Bildverarbeitungsalgorithmen in Vordergrund und Hintergrund separiert. Das Ergebnis wird in einer *Alpha-Matte* gespeichert und kann dazu verwendet werden, den Hintergrund mit einer anderen Videosequenz oder einer computergenerierten virtuellen Szene zu ersetzen. Während diese Techniken in vielen Produktionen eingesetzt werden, erzwingen sie die Verlagerung des Drehortes in ein virtuelles Studio und schränken die Kreativität der Filmemacher und Schauspieler durch die notwendigen blauen oder grünen Hintergründe und deren spezieller Ausleuchtung ein.

Eine Reihe von Forschungsprojekten verwendeten spezielle Beleuchtungen im Kontext von digitaler Videokomposition um die Keyingqualität von Produktionen in virtuellen Studios zu verbessern [163], um Moderationsinformationen wie Richtungsanweisungen anzuzeigen [63, 109] oder um

die Geometrie der Szene zu analysieren. All diese Anwendungen können jedoch nur in Kombination mit traditionellen Chromakeying-Systemen angewendet werden, daher haben sie die gleichen Einschränkungen und erfordern blaue bzw. grüne Hintergründe. Komplexere visuelle Effekte, wie zum Beispiel die Simulation von Lichtbrechungen transparenter Vordergrundobjekte werden meist aufgrund der fehlenden Brechungsinformationen in aufwändiger manueller Nachbearbeitung imitiert.
Zur Erzeugung perspektivisch korrekter Bildüberlagerungen wird die aktuelle Kameraposition benötigt, die mit verschiedenen Trackingverfahren ermittelt werden kann. Die Verwendung elektromagnetischer Trackingsysteme erfordert einen nicht unerheblichen zusätzlichen Hardware-und Kalibrierungsaufwand. Eine andere Möglichkeit bietet daher das optische Tracking, welches mit markerbasierten und markerlosen Techniken durchgeführt werden kann. Markerlose Verfahren erfordern eine bestimmte Menge an signifikanten Merkmalen in den aufgenommenen Bildern, während markerbasierte Techniken sichtbare Markierungen benötigen, die in der Aufnahmeumgebung montiert werden müssen. Diese Markierungen sollten jedoch weder direkt in der Studioumgebung noch in der aufgezeichneten Videosequenz sichtbar sein. Folglich ist markerbasiertes Tracking darauf angewiesen, Markierungen in unkritischen Bereichen, wie an der Decke oder auf dem Boden zu platzieren, die dann mit zusätzlicher Hardware erkannt werden können [159]. Während diese Verfahren durch Verdeckungsprobleme fehleranfällig sind, schränken sie die Anwendung auf vollständig kontrollierbare Studioumgebungen ein.

Die erwähnten, auch heutzutage noch in vielerlei Hinsicht eingeschränkten Möglichkeiten der digitalen Videokomposition motivierten uns, mit Hilfe von handelsüblichen Hardwarekomponenten neue Algorithmen und Methoden zu entwickeln, mit denen die Anforderungen zur Generierung digitaler Videokompositionseffekte vereinfacht werden können. Diese sollten zu flexiblen, einfach zu bedienenden Systemen kombiniert werden, mit denen die Probleme der traditionellen Verfahren gelöst und neue Möglichkeiten zur Generierung von visuellen Effekten ermöglicht werden können.

Ergebnisse dieser Arbeit

In dieser Arbeit werden drei unterschiedliche Ansätze präsentiert, die aufzeigen, wie man mit Hilfe von räumlich und zeitlich modulierter Beleuchtung die notwendigen Bedingungen zur Generierung von herkömmlichen visuellen Effekten für Film- und Fernsehproduktionen vereinfachen kann und mit diesen Techniken neue Möglichkeiten der digitalen Videokomposition realisiert. Die entwickelten Systeme werden ausführlich erörtert und es werden Anwendungsbeispiele gegeben, wie die

modulierte Beleuchtung dazu verwendet werden kann, verschiedene Effekte zur Videokomposition zu erzeugen, die mit den herkömmlichen Methoden, wie virtuellen Studios, nicht realisierbar sind. Während mit dieser Technik einerseits die in der Motivation angesprochenen Probleme gelöst werden können, stellt andererseits die Integration solcher Systeme in bestehende Aufnahmesysteme eine weitere Herausforderung dar.

Ziel dieser Arbeit ist es, hochfrequente, synchronisierte Beleuchtung, die sowohl räumlich als auch zeitlich in ihrer Intensität und Wellenlänge moduliert ist, dazu zu verwenden, um, neben der Szenenbeleuchtung, weitere technische Informationen welche zur Erzeugung von visuellen Effekten notwendig sind, darin zu integrieren. Dies wird so realisiert, dass die technischen Komponenten, die in die Beleuchtung integriert sind, das resultierende Video (das *Composite*) nicht beeinflussen und daher auch für die Betrachter nicht sichtbar sind. Mit diesem Ansatz wird versucht, die Einschränkungen des traditionellen Chromakey-Verfahrens und virtueller Studio-Produktionen zu beseitigen und darüberhinaus den Filmschaffenden neue, flexible Möglichkeiten zur Erzeugung von visuellen Effekten zu bieten.

Eine Anwendungsbeispiel ist in Kapitel 1.2, Bild 1.1 illustriert: Ein Fernsehstudio wird hierbei mit synchronisierten Hochgeschwindigkeits-LEDs und Projektoren beleuchtet. Diese projizieren modulierte Beleuchtungsmuster, welche kodierte Informationen beinhalten. Die in der Illustration dargestellten Codes sind in dem finalen Video nicht sichtbar, je nach Anwendungsfall auch nicht für die Personen im Studio. Sie können jedoch mit einer synchronisierten Kamera dazu verwendet werden, verschiedenartige Daten zu extrahieren, die zur Ermittlung von Verdeckungsinformationen oder der aktuellen Kameraposition dienen. Wie in Illustration 1.2 dargestellt, können diese für eine Reihe von visuellen Effekten für die digitale Videokomposition genutzt werden.

Die wissenschaftlichen Beiträge dieser Arbeit lassen sich wie folgt zusammenfassen:

- **Zeitmultiplexe, räumlich modulierte Beleuchtung**
  Entwicklung von synchronisierten Beleuchtungssystemen die sowohl eine räumliche als auch zeitliche Modulation der Szenenbeleuchtung ermöglichen, mit deren Hilfe Effekte der digitalen Videokomposition entweder in Echtzeit, oder zu Nachbearbeitungszwecken generiert werden können

- **Zeitmultiplexe Keyingmethoden**
  Implementierung und Auswertung verschiedener Keyingmethoden zur Generierung von hochqualitativen Alpha-Mattes

- **Optimierung des klassisches Chromakeying**

  Verwendung von synchronisierten, zeitlich variablen Hintergründen, um die Einschränkungen der herkömmlichen Chromakey-Techniken bezüglich Farbabhängigkeit und Colorspill zu überwinden

- **Adaptive Projektion nicht wahrgenommener Codes**

  Integration von unsichtbaren Mustern in projizierte Bilder, die zur nicht wahrgenommenen, kontinuierlichen Kalibrierung, zum Kameratracking oder zur Tiefenabtastung eingesetzt werden können

  - Δ-kodierte Projektion

    Echtzeitgenerierung der benötigten Bildpaare welche die unsichtbaren Codes beinhalten

  - Berechnung der Δ-Funktion in Echtzeit

    Ermittlung der dynamischen Parameter, die eine optimierte, nicht sichtbare Integration der Codes in die Bilddaten ermöglichen und dies auch unter kritischen Situationen, wie schnellen Augenbewegungen oder Codeänderungen garantieren

  - Beispielanwendungen für die entwickelten Algorithmen zur Generierung von dynamischen, nicht sichtbaren, projizierten Codes

    Implementierung eines synchronisierten Projektor-Kamera-Beleuchtungssystems welches die unsichtbaren Muster dazu verwendet, ein adaptives, optisches Kameratracking durchzuführen

  - Prüfung der ermittelten Funktionsparameter mit Hilfe von Nutzerstudien

    Validierung der Unsichtbarkeit der adaptiven, eingebetteten Codes durch Fallstudien

- **Live-Techniken zur digitalen Videokomposition**

  Implementierung eines echtzeitfähigen prototypischen *"Augmented Studio"*-Systems, welches mit Hilfe von nicht wahrgenommenen, sich dynamisch anpassenden, zeitlich und räumlich codierten Beleuchtungsinformationen die Generierung verschiedener Effekte zur digitalen Videokomposition, wie optisches Kameratracking und Flash Keying unterstützt

  - Alphamatting

    Verwendung von synchronisierten LED Beleuchtungseinheiten zur Erstellung von hochqualitativen Alpha-Mattes mit Hilfe von Flash Keying

  - Visuell nicht wahrgenommenes optisches Kameratracking
    Dynamische Anpassung der eingebetteten, unsichtbaren Codemuster um sie vollkommen vor der menschlichen Wahrnehmung zu verstecken, die Trackingmuster jedoch gleichzeitig durch eine synchronisierte Kamera mit Hilfe von dynamischer Rückkopplung robust extrahieren zu können

- **Techniken zur digitalen Videokomposition in der Postproduktion**
  Verwendung der synchronisierten Beleuchtungssysteme zur Generierung von visuellen Effekten in der Videonachbearbeitung

  - Alphamatting
    Luminanzkeying in beliebigen Umgebungen durch die Neutralisation der potentiell räumlich variierenden Reflexionseigenschaften der Projektionsoberfläche

  - Environmentmatting
    Projektion von horizontalen und vertikalen Farbverläufen zur Kodierung von Raumkoordinaten, um die Simulation von Lichtbrechungseffekten (Refraktionen) transparenter Vordergrundobjekte in beliebigen Umgebungen zu ermöglichen

  - Farbunabhängiges, temporales Chromakeying
    Erweiterung des traditionellen Chromakey-Verfahrens durch den Einsatz von aktiven, hochfrequent schaltbaren Hintergründen, die abwechselnd Komplementärfarben darstellen, um die Farbabhängigkeit zu vermeiden

  - Neutralisation von ungewolltem Colorspill
    Entfernung des ungewollten Colorspills der während des Chromakey-Verfahren entsteht, indem zwei aufeinanderfolgende Bilder mit jeweils komplementären Hintergrundfarben verarbeitet werden und so anstelle des Colorspills eine neutrale, weiße Hintergrundbeleuchtung simuliert wird

- **Implementierung der synchronisierten Hardwaresysteme**
  Anwendung von aktuellen Kamerakalibrierungstechniken und speziellen Hardwaremodulen, um die Aufnahmegeräte mit den Beleuchtungskomponenten, d.h. LEDs und Projektoren zu synchronisieren. Spezifikation der Hardwareanforderungen, mit denen eine präzise, zeitlich kontrollierte Beleuchtung erzielt werden kann

Während alle aufgelisteten Techniken dazu verwendet werden können, verschiedene visuelle Effekte zur digitalen Videokomposition zu generieren, können nicht alle gleichzeitig in einem System kombiniert werden. Da die Methoden teilweise für verschiedene Anwendungsgebiete konzipiert wurden, schließen sie sich gegenseitig aus. Aus diesem Grund werden im Rahmen dieser Arbeit drei verschiedene Systeme präsentiert, die alle auf dem gleichen Grundprinzip einer synchronisierten, zeitlich und räumlich kodierten Beleuchtung basieren, sich jedoch in ihren Möglichkeiten und dem Anwendungsziel unterscheiden. Neben der zugrundeliegenden Synchronisation aller Systemkomponenten bestand die Grundanforderung, dass handelsübliche Komponenten wie Firewire- oder Filmkameras und normale Projektoren zum Einsatz kommen, um den Kalibrierungs- und den finanziellen Aufwand zur Realisierung gering zu halten. Neben der Implementierung der Systeme und Anwendungsbeispiele im Bereich der digitalen Videokomposition, wie Alphamatting, Optimierungen für Chromakeying, Environmentmatting, optischem Kameratracking und der Darstellung von Moderationsinformationen, wurden Nutzerstudien durchgeführt, um zu messen, inwiefern die entwickelten, hochfrequenten Bildmodulationen die Betrachter beeinflussen. Im Folgenden werden die einzelnen Beiträge dieser Arbeit und die drei entwickelten Systeme näher erläutert.

Im Rahmen dieser Dissertation wurde unter anderem eine Echtzeit-Anwendung für die digitale Videokomposition entwickelt. Diese ermöglicht die Projektion von Bildern mit eingebetteten, sich dynamisch anpassenden Codes, die für das menschliche Auge nicht wahrnehmbar sind, aber von einer synchronisierten Kamera rekonstruiert werden können, was für optimiertes optisches Kameratracking genutzt werden kann. Ein synchronisiertes LED-Beleuchtungssystem wird zur Generierung von Alphamattes mittels eines intensitätsbasierten Differenzkeyings eingesetzt. Darüber hinaus wird auch beschrieben, wie die Synchronisation der einzelnen Komponenten bei einer Bildwiederholrate von 120 Hz erzielt wird und das System für ein Augmented Reality-Fernsehstudio verwendet werden kann.

Weiterhin wird gezeigt, wie mit Hilfe von radiometrischer Kompensation, einem Verfahren zur Neutralisation der inhomogenen Reflexionseigenschaften der Projektionsoberfläche, Farb- und Helligkeitsmuster auf beliebigen Oberflächen projiziert werden können, um zum Beispiel Bereiche der aufgenommenen Szene in Vordergrund- von Hintergrundbereiche zu trennen oder Moderationsinformationen darzustellen. Integriert man diese Technik in ein zur Kamera synchronisiertes, temporal moduliertes Beleuchtungssystem, können verschiedene visuelle Effekte zur digitalen Videokomposition auch außerhalb von virtuellen Studios ermöglicht werden. Durch das hochfrequente

Zeitmultiplexen der Projektion und der Szenenbeleuchtung kann sowohl hochwertiges Keying als auch Kameratracking in realen Umgebungen außerhalb von Studios durchgeführt werden, wobei diese Technik, im Gegensatz zu Produktionen in virtuellen Studios, die Verwendung des gesamten Szenenvorder- und Hintergrundes im finalen Composite erlaubt.

Mit denen in dieser Arbeit präsentierten Techniken ist es möglich, ein qualitativ hochwertiges, temporales Chromakey-Verfahren umzusetzen, mit dem die beiden erwähnten Probleme bezüglich Farbabhängigkeit und Colorspill durch die Verwendung von zwei zueinander komplementär gefärbten, aktiven Hintergründen behoben werden können.

Neben der softwareseitigen Implementierung der entwickelten Algorithmen wurden im Rahmen dieser Arbeit auch angepasste Hardwaremodule entwickelt, mit denen eine frei konfigurierbare Synchronisation der verschiedenen Systemkomponenten mit einer Taktrate von bis zu 120 Hz durchgeführt wurde. Dies umfasste die Entwicklung von mikrocontrollergesteuerten, dimmbaren LED Beleuchtungseinheiten, Signalgeneratoren und -empfängern, mit denen Beleuchtung, Projektoren und die Kamera synchronisiert wurden.

ACKNOWLEDGMENTS

First of all, I would like to thank my supervisor Prof. Oliver Bimber for his support, guidance and continuous assistance throughout the years of my Diploma and Ph.D studies. I would also like to thank Prof. Mark Billinghurst for kindly taking the time to be my second supervisor.

In addition I am grateful to the German Research Foundation (DFG) who funded work on two different research projects that would not have been possible without their financial support.
I would also like to thank Gordon Wetzstein and Sebastian Knödel for all the great years in Weimar and our mutual stimulation to do computer graphics research which encouraged me to start my doctoral studies in this field of research. Of course, I am also thankful to my colleagues Erich Bruns and Chrystoph Toll for all the motivational support over the last few years, and to Hans Pabst and Jan Springer for the C++ programming skills I learned from them. And, of course, thanks to all my Hiwis, Max Grosse, Sebastian Thiele, Jan-Martin Strehlow, David Exner, Thomas Klemmer and Pilar Andrea Morales without whom my work would not have been possible. In this context I also want to thank Ferry Häntsch most sincerely for all the incredible work he did to make all my systems work. Furthermore I would also like to thank Daniel Thompson, Katharina Sengstaken and Cameron for their nice audio recordings, and especially Julian Reisenberger for proofreading my thesis. I am also grateful my parents and Thommy, Tim, Steppo, Stefanie, Tobias, Alex and all the others who supported me over the last few years in various ways!

Finally, for her great support, encouragement and patience over all the years, I would like to thank Kristin.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

The communication of information and story telling in particular is one of the basic needs of mankind. While in earlier millennia murals and later paintings, handwriting and letterpress printing were used as a means of communicating, humans also began to draw image sequences, which, if presented in rapid succession, generated the impression of an animated sequence. In its simplest form this can be achieved with a flip book, but automated systems were also developed that used special rotating image plates in combination with slit apertures, mirrors or optics – the so-called phenakistiscopes, zoetropes or praxinoscopes.

With the invention of photography, people such as Eadweard Muybridge, Etienne-Jules Marey and Ottomar Anschütz quickly began to record image sequences and to replay them in quick succession – as *film*. With the advent of professional movie production the first attempts were soon undertaken to generate special visual effects to increase the degree of immersion by using stop-motion puppetry, double exposures or background paintings [138]. While the visual quality of these effects was relatively poor in the era of analog film production until the 1980s, their use and quality rapidly increased with the rising technical capabilities of the semiconductor and the resulting available processing power and simplified digital editing possibilities. The enormous opportunities presented by lossless post-processing in combination with photorealistic three-dimensional renderings has given rise to the widespread use of a variety of visual effects in movies in general. An examination of the success of current cinema productions shows that a majority of them have also won *academy awards* for visual effects [115, 144]. This confirms the fact that the popularity as well as the lucrativeness of movie productions seems, among other things, to be closely related to the application of high-quality professional visual effects.

Although visual effects have been used for decades in digital film and television production, they still require an enormous amount of manual processing and are limited in their flexibility because of the techniques used. Each professionally created visual effect may use established techniques, such as chroma keying, to calculate the necessary image information, but still needs a huge amount of manual post-processing to render the final composite in a realistic way. While chroma keying has become the de-facto standard for creating alpha mattes which are the basis for most visual effects, the method has limitations with respect to color dependency and color spill. Since the separation of foreground and background is calculated depending on the pixel's chromatic value, it is obvious that foreground objects need to have a dissimilar color to the background or else they will

be assigned incorrectly to the background. If color similarities cannot be avoided, time consuming manual correction during post-processing will be necessary [143]. Color spill denotes the unintentional presence of the background color used for keying in the final composite. This occurs in semi-transparent regions, such as hair, but also due to indirect reflections of the background color onto foreground objects.

Even today, the highly automated generation of high-quality visual effects in environments that cannot be perfectly controlled such as virtual studios, remains one of the foremost problems in digital video composition and usually requires a considerable amount of manual work: During recording, parts of the background are covered with mobile blue or green backdrops and the scene may be recorded several times with automated, repeatable camera movements. Clearly this constrains the creativity of such recordings. During post-production, the missing parts of the scene background have to be seamlessly replaced by other video sequences, increasing the total production costs considerably. The absence of the virtual components of the final composite, i.e. the background areas not present when filming, also requires well-trained performers or moderators who can act naturally in such environments [162].

The limited temporal resolution of human visual perception is exploited in film to create the impression of continuous motion from a series of still images. Other techniques, such as zoetropes, or the projection-based Morphovision Display [59] exploit the limited temporal resolution of the human eye to create animations, for example, by illuminating static objects with stroboscopic light flashes. By analyzing the current viewing conditions, this temporal restriction can be used to hide certain information, encoded in modulated light, from the human eye. A synchronized camera, however, can be used to reconstruct the hidden information, which can be used as the basis for generating visual effects, while remaining hidden to the observer. As the visibility of this hidden information is influenced by a number of factors, it has not yet been possible to make hidden information complete invisible.

The central intention of this dissertation is identify ways of overcoming the classical constraints and problems of digital video compositing by using specific, temporally and spatially modulated illumination patterns synchronized to the recording camera. In what follows we will describe a series of flexible solutions that show how such systems can be used to simplify the generation of visual effects and how they can be applied to offer completely new possibilities for digital video compositing.

1.1 Motivation

Besides home entertainment and business presentations, video projectors are powerful tools for modulating images spatially as well as temporally. The new re-evolving need for stereoscopic displays increases the demand for high-frequency projectors with low latencies. Recent advances in LED technology also offer high modulation frequencies. Combining such high-frequency illumination modules with synchronized, fast camera sensors, makes it possible to develop specialized high-speed recording systems for visual effect production, such as, for example, the light stage [44], which enables the generation of image-based re-illumination effects within a completely controlled illumination environment. In combination with the increasing parallel processing power of modern graphics devices and the resulting potential for real-time image processing, it becomes possible to generate real-time composites using such synchronized illumination systems.

Today, many film and television productions apply visual effects by using virtual studio technology. The main method used to generate the required separation of foreground and background regions is chroma keying, which requires a homogeneously lit blue or green background surface. The recorded video signal then is analyzed either on-line or with post-production software by using image processing algorithms to separate the foreground from background regions. The result is stored in a so-called alpha matte and can be used to replace the background with another video sequence or a rendered virtual scenery. While these techniques are well established, they, constrain the recording environment to the virtual studio location and, due to the required physical blue or green backdrops, restrict the creativity of filmmakers and actors. A series of research projects applied specialized illumination within the scope of digital video compositing to enhance the keying quality in the context of virtual studio productions [163], to display moderator information, such as direction instructions [63, 109] or to analyze the scene geometry. All these applications, however, can only be applied in combination with traditional chroma keying systems and thus require physical blue or green screens and suffer from the same limitations. More complex visual effects, such as the simulation of refractions in transparent foreground objects are mostly faked using manual post-processing because of the missing refraction information. The generation of perspectively correct augmentations requires knowledge of the current camera pose which can be obtained by different camera tracking techniques, e.g. electromagnetically or through optical tracking. While the former requires additional hardware and calibration the latter can be carried out with marker-based or marker-less methods. While the marker-less techniques require a certain amount of detectable features in the recorded images, marker-based methods require visible markers to be mounted some-

where in the recording environment, but these markers should neither be directly visible within the studio environment nor appear in the recorded video stream. Consequently, marker-based tracking is mostly restricted to observing out-of-shot areas, such as the ceiling or the floor, using additional hardware [159]. While this is error prone, it also restricts the application to completely controllable studio environments.

The need to overcome the limitations of the above means of digital video composition and the variety of constraints involved provided the initial motivation of this thesis. By using new methods of synchronized illumination in combination with standard hardware, the aim is to simplify the generation of visual effects. The primary idea focuses on investigating and implementing flexible hardware systems, consisting of video projectors, LED lighting and cameras, to generate and record synchronized, spatially as well as temporally modulated illumination patterns. These can be used to overcome the existing problems of traditional techniques and to open up new possibilities in the context of digital video composition for film and television productions.

1.2 Contribution

In this thesis we present different approaches for using spatially as well as temporally modulated illumination in combination with a synchronized camera system to simplify the requirements of standard digital video composition techniques for film and television productions and also to offer new possibilities and overcome the limitations of such kinds of systems. A discussion along with applications are shown of how modulated light can be applied to a scene recording context to enable a variety of digital video compositing effects which cannot be realized using standard methods, such as the virtual studio technology. While this approach solves several of the issues mentioned in the motivation section above, it also presents a series of new challenges of how to integrate them into a recording environment.

We propose using high-frequency, synchronized illumination which is modulated spatially as well as temporally in terms of intensity and wavelength to, in addition to the scene illumination, encode technical information for visual effects generation. This is carried out in such a way that the technical components do not influence the final composite and are thus not visible to the viewers. Using this approach we intend to overcome the constraints of traditional chroma keying and virtual studio productions and to generate new flexible possibilities for filmmakers to overcome the limitations

Figure 1.1: Example scene with visualized coded projection

of conventional methods. [1]

An example application scenario is shown in the illustration in figure 1.1: a studio environment is illuminated by synchronized high-frequency LEDs and projectors project code patterns which are used to encode modulated information spatially as well as temporally. The codes visualized in the illustration are not visible in the resulting final composite. Depending on the application scenario and the requirements, the coded information embedded in the illumination is also not visible to the people within the studio environment. It can, however, be used to acquire a number of different kinds of data such as occlusion information and the current camera pose which can be used to generate a variety of visual effects in the final composite as illustrated in figure 1.2.

To summarize, the main contributions of this thesis can be described as follows:

- **Time-multiplexed spatial illumination**
  Development of synchronized systems to generate a spatially as well as temporally controllable scene illuminations for the generation of on-line and off-line digital video composition effects

[1] Some of the basic ideas realized in the context of this thesis were summarized in [15].

Figure 1.2: Example scene of our main idea as presented to the final audience. The codes shown are not visible in the final composites and are used to generate perspectively correct augmentations.

- **Time-multiplexed keying methods**
  Implementation and evaluation of various keying techniques for generating high-quality alpha mattes
- **Chroma keying optimizations methods**
  Use of synchronized temporal backdrops to overcome the limitations of standard chroma keying in terms of color dependency as well as color spill
- **Adaptive imperceptible pattern projection**
  Integration of non-visible patterns into projected imagery that can be used to support continuous on-line-calibration, camera tracking, and the acquisition of scene depth
  - Δ-coded projection
    Real-time generation of image pairs containing imperceptible code patterns
  - Estimation of the parameters required to calculate the Δ-function
    A user study evaluation of the influence of various parameters to estimate their relationship for the definition of an adaptive function for imperceptible code embedding

  - Calculation of the Δ-function

    Real-time calculation of dynamically adaptive parameters for optimized imperceptible code integration that cannot be seen even during fast eye movements as well as code alternation

  - Application of dynamic codes

    Implementation of a synchronized projector, camera and illumination system dynamically adapting the imperceptible codes

  - Validation using a user study

    Verification of the imperceptibility of the developed adaptive coding strategies using a user study

- **On-line digital video composition techniques**

  Implementation of a real-time, proof-of-concept augmented studio prototype supporting instant digital video compositing using imperceptible flash keying as well as optical camera tracking and dynamic code adaptation

  - Alpha matting

    Use of synchronized LED illumination units to generate high-quality alpha mattes using flash keying

  - Visually imperceptible optical camera tracking

    Adapting tracking patterns invisible to humans, but optimized for camera visibility using dynamic feedback processing

- **Off-line digital video composition techniques**

  Use of synchronized illumination systems to generate various digital video composition effects during post-processing

  - Alpha matting

    Projection-based temporal luma keying in arbitrary environments by neutralizing the underlying reflectance properties of the projection surfaces

  - Environment matting

    Projection of colored ramps to encode spatial coordinates to support the simulation of refractions using environment matting in arbitrary environments

  - Color-independent chroma keying
    Enhancing classical chroma keying by using temporally based complementary colors as backdrops to overcome the traditional problem of color dependency
  - Color spill neutralization
    Neutralization of undesired color spill that occurs during chroma keying recordings by combining consecutively recorded imagery with complementary backdrop colors

- **Implementation of synchronized hardware setups**
  Application of state-of-the-art camera calibration and synchronization of the capture devices with the illumination devices. Specification of hardware requirements for generating controllable synchronized LED as well as projector illumination

While all of these techniques can be used to generate various effects for digital video composition in film and television productions, not all of them can be combined simultaneously in one system. This is because some of the different methods aim to resolve application scenarios which are mutually exclusive. Accordingly, in this thesis we present different realizations of synchronized, temporal and spatial illumination systems, each serving particular needs. Besides the synchronization of all system components, the systems were developed with standard components only, such as firewire or studio cameras and off-the-shelf video projectors to minimize the financial as well as the calibration needs. In addition to the development of various applications for digital video compositing, such as alpha matting, chroma keying optimizations, environment matting, imperceptible optical camera tracking and in-scene moderation support, user studies were also carried out to analyze if the developed high-frequency temporal image modulations might influence observers.

Besides the software implementation, specialized hardware components were developed to guarantee the configurable synchronization of the different components at the speed of up to 120 Hz. Specifically, a custom-built, dimmable LED illumination system with associated control units and the generation and distribution of synchronization signals for the camera and the projectors had to be developed.

1.3 Outline

The following chapters of this thesis are structured as follows:

Chapter 2 introduces the basic terminology, details necessary background information and summarizes previous as well as related work in the field of digital video compositing, imperceptible coded projections, applications for projector-camera systems and optical marker tracking. The section summarizes and classifies the wide field of keying techniques and the individual advantages as well as disadvantages of the respective methods and goes on to discuss approaches more closely related to our own approach in more details. These methods focus on time multiplexed, digitally controlled studio illumination techniques which are used to generate different special effects for on-line or off-line video post-production. The different advantages of our proposed solutions will also be discussed and compared to the previous methods mentioned in this section.

An recapitulating explanation of the different contributions of this thesis is given in chapter 3. We present a series of applications for synchronized, spatially and temporally controlled and modulated illumination systems in the context of digital video compositing. We show, how different keying methods, such as flash keying, luma keying and chroma keying can be carried out efficiently with such kinds of systems. In addition, a method is presented which efficiently neutralizes the undesired color spill in the final composites. Furthermore, a variety of other applications besides keying, such as optical tracking, geometric calibration and radiometric compensation, adaptive imperceptible coded projection, the embedding of moderator- and camera-related information as well as environment matting are also discussed.

After giving an overview of the different methods developed, a real-time application for live digital video compositing is described in detail in chapter 4. It supports the projection of animated imagery containing dynamically adapting codes which are imperceptible to human eyes but clearly visible to the camera, enabling real-time optical tracking. The synchronized LED lighting system facilitates intensity-based difference keying to support the generation of alpha mattes. In addition, we also describe the synchronization of the different components at a frame rate of 120 Hz as well as the OpenGL-based real-time implementation of a prototypical augmented television studio application.

In chapter 5 we focus on developing an illumination system for the generation of high-quality digital video composition effects in arbitrary environments through the off-line post-processing of

recorded video data. Again, the system we present provides fully synchronized LED and projector-based illumination, which makes it possible to project spatial and temporal information in a frame-precise manner, geometrically corrected and radiometrically compensated onto arbitrary surfaces. Such codes can be used during runtime, for example, to aid the interaction of the actors with non-visible parts of the final composite, or to enable special effects during off-line post-processing, such as camera path reconstruction via match-moving and compositing using alpha matting or environment matting.

A third system is presented in chapter 6 which enables color-independent chroma keying by alternately displaying green and magenta colors as backdrops at recording rate. While this overcomes the limitations of color dependency with the aid of a high-frequency, time-sequential technique, we also present a method for the real-time correction of errors resulting from this temporal approach. We present a solution on how to integrate the algorithms into existing digital video compositing pipelines and show how the approach can be extended to support camera path reconstruction. Furthermore, this approach also makes it possible to neutralize unwanted color spill visible in the final composites.

The synchronization of the individual hardware components and the custom-built control units, as well as the necessary response calibration of the individual components is explained in detail in chapter 7.

Finally, chapter 8 contains a summary of the work presented in this thesis along with conclusions and possible avenues of future research.

1.4 Publications

1.4.1 PhD related

Over the last five years of research, various aspects related to this dissertation have been published in journals and/or presented at different international conferences. The following list summarizes these publications. Those highlighted were published during the PhD phase.

Journal Contributions

O. Bimber, A. Grundhöfer, S. Zollmann and D. Kolster (Bauhaus-Universität Weimar). Digital Illumination for Augmented Studios. Journal of Virtual Reality and Broadcasting, 2006

A. Grundhöfer, O. Bimber (Bauhaus-Universität Weimar). VirtualStudio2Go: Digital Video Composition for Real Environments. ACM Transactions on Graphics (Siggraph Asia), 2008, Singapore

A. Grundhöfer (Bauhaus-Universität Weimar), D. Kurz (Bauhaus-Universität Weimar), S. Thiele (Bauhaus-Universität Weimar), O. Bimber (Johannes-Kepler-Universität Linz). Color Invariant Chroma Keying and Color Spill Neutralization for Dynamic Scenes and Cameras. Springer, The Visual Computer (Computer Graphics International), 2010, Singapore

A. Grundhöfer and O. Bimber (Bauhaus-Universität Weimar). Real-Time Adaptive Radiometric Compensation. IEEE Transactions on Visualization and Computer Graphics, Volume 14 Issue 1, 2008

Conference Contributions

O. Bimber (Bauhaus-Universität Weimar), D. Iwai (Osaka University), G. Wetzstein (University of British Columbia), A. Grundhöfer (Bauhaus-Universität Weimar). The Visual Computing of Projector-Camera System, Eurographics State of the Art Report, 2007, Praque, Czech Republic

A. Grundhöfer, M. Seeger, F. Häntsch, O. Bimber (Bauhaus-Universität Weimar). Dynamic Adaptation of Projected Imperceptible Codes. International Symposium on Mixed and Augmented Reality (ISMAR), 2007, Nara, Japan

A. Grundhöfer, O. Bimber (Bauhaus-Universität Weimar). Dynamic Bluescreens. Siggraph Research Poster and Talk, 2008, Los Angeles, USA

This thesis was partially funded by the DFG project as part of the *Augmented Studio* project, reference number BI 835/2-1.

1.4.2 Further Publications

Furthermore, Anselm Grundhöfer contributed to the following scientific publications not directly related to this dissertation during his PhD studies:

A. Grundhöfer, O. Bimber (Bauhaus-Universität Weimar), Real-Time Adaptive Radiometric Compensation, ACM Student research competition (SRC), 2007, San Diego, USA

F. Petzold, O. Bimber, C. Tonn, A. Grundhöfer, D. Donath (Bauhaus-Universität Weimar), Spatial Augmented Reality for Architecture - Designing and planning with and within existing buildings, International Journal of Architectural Computing, 2008

O. Bimber (Bauhaus-Universität Weimar), D. Iwai (Osaka University), G. Wetzstein (University of British Columbia), A. Grundhöfer (Bauhaus-Universität Weimar), The Visual Computing of Projector-Camera System, Computer Graphics Forum, 2008

M. Grosse (Bauhaus-Universität Weimar), G. Wetzstein (University of British Columbia), A. Grundhöfer (Bauhaus-Universität Weimar), O. Bimber (Bauhaus-Universität Weimar), Adaptive Coded Aperture Projection, ACM Siggraph Talk and Poster, 2009

M. Grosse (Bauhaus-Universität Weimar), G. Wetzstein (University of British Columbia), A. Grundhöfer (Bauhaus-Universität Weimar), O. Bimber (Bauhaus-Universität Weimar), Coded Aperture Projection, ACM Transactions on Graphics, 2009 (Presented at ACM Siggraph 2010)

O. Bimber (Bauhaus-Universität Weimar), Toshiyuki Amano (Nara Institute of Science and Technology), A. Grundhöfer (Bauhaus-Universität Weimar), D. Kurz (Bauhaus-Universität Weimar), F. Häntsch (Bauhaus-Universität Weimar), S. Thiele (Bauhaus-Universität Weimar), Projected Light Microscopy, ACM Siggraph Talk, 2009

D. Exner (Bauhaus-Universität Weimar), E. Bruns (Bauhaus-Universität Weimar), D. Kurz (Bauhaus-Universität Weimar), A. Grundhöfer (Bauhaus-Universität Weimar), and O. Bimber (Johannes-Kepler-Universität Linz), Fast and Robust CAMShift Tracking, The First IEEE International Workshop on Computer Vision for Computer Games, 2010

O. Bimber (Bauhaus-Universität Weimar), Toshiyuki Amano (Nara Institute of Science and Technology), A. Grundhöfer (Bauhaus-Universität Weimar), D. Kurz (Bauhaus-Universität Weimar), F. Häntsch (Bauhaus-Universität Weimar), S. Thiele (Bauhaus-Universität Weimar), Closed-Loop Feedback Illumination for Optical Inverse Tone-Mapping in Light Microscopy, IEEE Transactions on Visualization and Computer Graphics (submitted '09, published '10)

2 Background and Related Work

The outcome of this thesis consists of a variety of methods for digital video composition utilizing techniques originating from different fields of research. While all the systems make use of the same basic hardware components – synchronized cameras, projectors, LED illumination units as well as graphics cards – the developed applications are based on research originating from the fields of computer vision, image processing, digital video composition, human visual perception as well as real-time rendering. In the following sections of this chapter, the basic terms of the techniques used will be introduced and discussed and previous work related to the methods presented in this thesis will be summarized.

Firstly, in section 2.1.1 we will introduce the basic methods of digital video compositing applied in the context of all the systems developed in this thesis. This is followed by a general introduction to the human visual system and a detailed discussion of its temporal behavior in 2.1.2. It is these specific constraints that are used to hide projected technical information required for different applications from the human eye. The basic steps for generating a calibrated projector-camera system are explained in 2.1.3.
The related work summarized in 2.2 is divided into the following parts: in section 2.2.1 we provide a thorough overview and classification of the existing keying methods and how they are related to our approaches and discuss and compare their respective advantages and disadvantages. In section 2.2.2 we present different methods for reducing color spill, which is an undesired result of chroma keying, and in 2.2.3, we summarize various advanced applications using projector-camera systems in the context of digital video composition as well as to project code patterns imperceptible to the human eye. Finally in section 2.2.4, we discuss different related approaches for marker-based as well as marker-less optical tracking techniques.

2.1 Basic Terms and Concepts

This section introduces the basic terminology and concepts which will be used frequently in the following chapters of this thesis and provides a brief introduction to the basics of digital video compositing, and matting; in particular. We will give a general overview of the temporal behavior of the human visual system and its constraints and describe the basic methods for calibrating a projector-camera system geometrically as well as to compensate for inhomogeneous reflections from colored or textured surfaces.

2.1.1 The Basics of Digital Compositing

The primary objective of this thesis is the development of new possibilities for digital video composition by applying synchronized, spatially and temporally modulated projector-camera-illumination systems. All of the solutions presented make use of the same basic principals of video composition which will be discussed in the following.

The combination of multiple images or video sequences seamlessly into a new *composite* is one of the fundamental operations for generating visual effects for films and television productions. The process consists of two main steps: during the first step, so-called *keying* or *matting*, the input scene has to be classified into foreground and background regions. In the second step, *compositing*, the foreground region, defined by the *alpha matte* is placed over a new background. While the early approaches to compositing were lacking in terms of visual quality, visual effects production has become a huge and lucrative market over the years and – due to the exponential growth of processing power as well as the development of advanced algorithms – is able to produce high-quality video composites in which the different parts of the image are no longer distinguishable from one another, even by professionals. As mentioned in [34], the potential of digital video compositing is underlined by the fact that the majority of the ten best-selling movies of all time also won the Academy Awards for their visual effects. Besides its usage for film productions, compositing is now a standard method for image manipulations, advertising, as well as interactive media systems. Although the processing steps required to generate mattes as well as composites were developed decades ago, it is still a field of ongoing research: while a long list of methods exist to generate high-quality mattes, all currently available methods have some kind of restriction and cannot reproduce all physical effects in every situation. We will discuss the different matting methods in detail later in the related work section of this chapter and first focus in the compositing operation.
The digital compositing process is summarized by the *compositing equation* which was introduced by Porter and Duff [129]

$$C = \alpha F + (1 - \alpha) B \tag{2.1}$$

where C is the composite image, F and B foreground and background images, respectively and α is the alpha matte which defines an opacity value for each image pixel. This value is used to linearly blend the foreground and background contributions of each image pixel. Besides the operation shown in equation 2.1, other compositing operations were introduced in [129] which can be used to generate various kinds of composites. The *over* operation shown in equation 2.1, however, is the most common one.

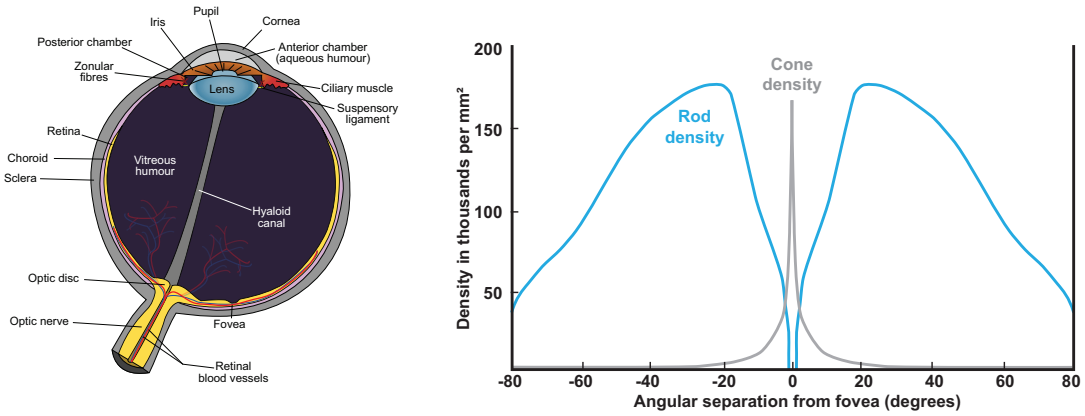
While the compositing step is a simple and completely defined operation, the calculation of the unknown α-value, the *alpha matte*, is a far more complex problem: it is actually the inverse operation to compositing and thus it is heavily under-constrained. While C is the known input image, the values for F, B and α are unknown. Thus, for a three color channel RGB image, a problem has to be solved containing three equations (one for each color channel), but with seven unknowns (the RGB components of F and B as well as α). Various methods have been proposed to overcome this problem by using multiple images, constrained backgrounds, specialized recording equipment, such as IR lighting, or multiple cameras and manual interaction in combination with statistical mathematical methods, etc. An overview of the various methods for solving this problem will be given in section 2.2.1.

2.1.2 Fundamentals of Temporal Vision

Projectors have been used in the context of this thesis for various applications, one of which is to hide information from human visual perception by encoding it in a temporal manner. This approach exploits the limited temporal resolution of the human visual system (*HVS*) to make the information visually disappear to the eye. A comprehensive overview over the HVS would exceed the scope of this dissertation, so instead, after a brief introduction to the human eye, we will focus on the basic principals of temporal human visual perception taken into account in the context of this thesis.

2.1.2.1 The Human Eye

The human eye is one of the most complex organs nature has evolved. Its main purpose is the generation of a visual impression from a narrow band of electromagnetic waves ranging from $380-780$ nm [173] surrounding us which we usually refer to as ”light”. Figure 2.1(a) shows an overview of a human eye, whose primary components can be briefly summarized as follows: the incoming light rays – more precisely: *photons* – pass through the cornea. The rays then pass the anterior chamber, the iris, the lens as well as the posterior chamber. The iris can be expanded or contracted by the ciliary muscle to adapt the light throughput depending on the amount of incoming light similar to a camera’s aperture. In combination with the anterior chamber as well as the cornea, the lens serves as a convergent lens to project the incoming illumination information onto the retina. The retina is populated with a huge amount of light-sensitive cells, the *photoreceptors* transforming the incoming illumination stimulation into neuronal impulses. They can be mainly classified into



(a) (b)

Figure 2.1: (a): Schematic overview over the human eye (source: Wikipedia [178]).
(b): Distribution of rods and cones on the retina (the blind spot, located either 15° left or right from the fovea at the position of the optic disc, is not illustrated in the diagram).

highly sensitive, achromatic *rods* [2] serving as detectors for brightness during low lighting conditions [3], and three kinds of much less sensitive, trichromatic *cones* [4] which are able to generate the impression of color [5]. This is achieved by the fact that all three kinds of cones differ in their spectral absorption characteristics and, consequently, react with different sensitivities to varying wavelengths. They are typically referred to as S-,M- and L-cones which are abbreviations for short, medium, and long wavelength sensitive cones [147]. The distribution of the photoreceptors over the retina is highly inhomogeneous as shown in figure 2.1(b) – the majority of all cones are placed in the central region of the retina, the *fovea*, which is responsible for sharp central vision, while the maximum number of rods are placed 10 – 20 degrees besides the eye's fixation point which gives this region the highest sensitivity to dim lighting conditions. This inhomogeneous distribution leads to varying visual impressions depending on the eye's fixation point and its periphery, which also influences the ability to discriminate high frequency effects as described in the next subsection. The neuronal impulses of the photoreceptors are highly compressed by other components of the retina and then further transported into the visual cortex of the human brain by the optic nerve to finally generate a visual impression.

[2] Approx. 120 millions per eye

[3] This state of visual perception is referred to as *scotopic vision*

[4] Approx. 6 millions per eye

[5] Visual perception under bright lighting conditions is referred to as *photopic vision*. *Mesopic vision* describes lighting conditions in which rods as well as cones are active.

The complete process of converting light into electric signals which can be transmitted by the nervous system is referred to as *visual phototransduction*. This highly complex chemical process is carried out within the photoreceptors and requires a certain amount of processing time as well as energy. This, as well as other processing tasks limits the temporal resolution of human visual processing [150] which will be discussed below.

2.1.2.2 Temporal Summation

The human eye continuously integrates the photons stimulating the photoreceptors over time and the incoming visual information projected onto the retina is perceived as a continuous flux. Depending on the stimulated photoreceptors (rods or cones) and the luminance (i.e. number of photons approaching), the HVS requires a minimum amount of time to process the incoming information. This minimum integration time limits the ability to distinguish between consecutive temporal variations. Depending on the illumination conditions, the average integration time can vary between 100 ms for scotopic vision and $10-15$ ms for photopic vision. During the integration time, a *temporal summation* takes place over a certain amount of time, called the *critical period* until the summation threshold is reached [91]. The correlation between the luminance, the stimulus duration as well as the total luminous energy in the critical period is described by *Bloch's law* [83] which states that if the luminance of a stimulus is halved, a doubling in stimulus duration is required to reach the same level of perception. Note that for stimuli present for longer than the critical period, this law no longer affects the perceived impression and a constant intensity is perceived independent of the stimulation time.

2.1.2.3 Temporal Flicker Sensitivity

The temporal resolution of the HVS is constrained by the *critical flicker frequency (cff)* which describes the frequency boundary at which a pulsating light source is no longer recognized as discrete flashes, but as a continuous flux of light. This means that all impressions presented above the cff are perceived as a continuously blended temporal summation as described above in 2.1.2.2. This effect is well-known from everyday life, for example, in movie presentations which present flashing still images at a frame rate above the cff so that they appear as a continuous animation. The cff is not a constant value, but depends on a list of factors, such as luminous intensity, stimuli size as well as spatial frequency of the stimuli. The relationship between the cff and the luminance of the flickering stimulus is described by the Ferry-Porter Law [26] which defines cff$= a \log L + b$ where

a and *b* are constants and *L* equals the luminance of the flashing stimuli. This law implies that the perceptibility of flicker increases with the amount of luminance. While the two constants can be adapted depending on the size of the stimulus and its projected position on the retina, it is still a simplified model, but has proved to be stable over a wide range of luminance intensities. The size as well as the position of the stimulus influences the cff because of the fact that rods and all three types of cones have different temporal resolutions and are not homogeneously distributed over the retina as described in 2.1.2.1.

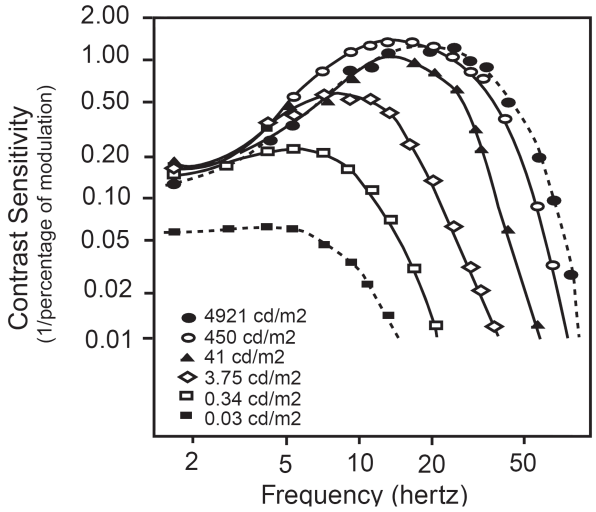


Figure 2.2: The temporal sensitivity to flicker with respect to stimuli contrast plotted for different luminance levels (source: [22]))

More complex temporal contrast sensitivity functions were developed [19] which also consider other parameters, such as the spatial frequency, flicker waveform, the target size, the contrast between background and stimuli intensity and the type of surround, the observers current light adaptation, their foveal or peripheral viewing, as well as sex and age of the subject. In general, however, the major factor influencing flicker perception is the luminance of the presented stimuli as shown in figure 2.2. The higher the curves, the easier it is to perceive flickering. Nevertheless, other factors such as, for example, the spatial frequency also play an important role [19].

Besides the aforementioned effects that influence the temporal sensitivity of the HVS, the human eye tends to continuously move very quickly which also has an influence on the perception of flickering targets. The different types of eye movements and the resulting effects are discussed in the next section.

2.1.2.4 Effects During Unintended Fast Eye Movement

Besides intentional eye movements to orient the gaze direction so that the desired objects are mapped onto the fovea, the human eyes move continuously and very quickly. Such fixational movements can be classified into three main categories: *Tremor*, which are barely measurable eye movements assumed to be necessary to keep the early visual system continuously active, *Drift*, which are required to compensate inaccurate motions resulting from *Microsaccades* which are also assumed to be necessary in order to maintain continuous stimulation to the retina's photo receptors. In addition, those fast motions are necessary to compensate for sudden eye movements, *saccades*, during a change in gazing direction from one fixation point to another as well as to stabilize the current fixation point. This means that the human eye is in continuous motion even during a perceived constant gaze direction.

While no intentional fixation is possible during the saccade and *saccadic suppression* leads to virtual blindness during that time [104, 185], humans can still perceive intensity variations: If a flickering stimulus, like a stroboscopic point light source, is presented during a saccade or fixational eye movement, a number of bright dots, a so-called *phantom array* [75, 76], is perceived. Note that such intensity variations also become apparent during slight illumination changes. They result from the fact that during eye movement, the photons of the light source are projected onto slightly different portions of the retina. Similar, humans perceive an illuminated line (*phantom line*) [116] if a continuous point light source is presented. These effects can also be observed when watching a DLP-based projection: the different colors of the color wheel become apparent at high intensity edges of the projection during eye movements. The effect is known as the *color flash effect*, *color rainbow effect* or *color fringing* [22].

In addition to these undesired effects, the perception of the *phantom array* can be used to generate 2D displays by using a 1D array of flickering light sources moving at a high speed [172, 3], or hiding images within projections that are only visible during saccadic eye movement [105]. Other groups have used the limited temporal resolution of the human eye to enhance the perceived resolution of displayed images by displaying moving content in a frequency above the cff [47]. Further studies reveal that even in situations where high frequency illumination above the critical flicker frequency is not perceived, it can influence the general task performance of the human brain [81].

We utilized the mentioned temporal constrains of the HVS to hide technical information required for digital video compositing in the illuminating components of the developed systems by embed-

ding them with adaptively changing parameters at a frequency clearly above the cff. This will be described in detail in chapter 4.

2.1.3 Projector–Camera Systems

To generate the digital video compositing effects described in this thesis, cameras are used in combination with LED illumination and one or multiple projectors to display spatially as well as temporally modulated illumination patterns which encode the required information. Depending on the application, projectors were used not only to project onto optimized projection screens, but also to display encoded information on arbitrary surfaces which may be geometrically distorted and/or textured. For this, the surface's geometric structure and its reflectance properties have to be measured in advance to neutralize their influence on the projected coded information. The following section explains the registration and compensation steps required to project a geometrically and radiometrically undistorted image.

2.1.3.1 Geometric Registration

As already summarized in the introduction, the main goal of this thesis is the use of temporally as well as spatially controllable illumination for digital video composition. While a temporal adjustment of the individual components can be achieved by synchronizing all components to a reference clock, an initial calibration step has to be carried out to estimate the basic relationships between the individual components to generate spatially controllable illumination patterns. Depending on the projection surface, geometric warps such as keystone distortions in the case of slanted projections or even more complex effects might occur, as, for example, in the case of projecting onto a non-trivial geometry such as a natural stone wall. Such kinds of distortions have to be analyzed and corrected to enable a precise spatially controllable and undistorted projection. For this, a link between the projected pixel coordinates and the projection surface has to be established to control the projected intensities on a per-pixel basis. Depending on the application, different methods can be applied to achieve this goal which will be summarized below [6].

If the projection surface is planar the mathematical concept of *homographies* can be used to generate a relationship between the projection image and the surface. In computer vision a homography is defined as the projective mapping of points from one plane to another and can be expressed by

[6] A detailed overview of the various methods is given in [16]

a 3x3 orthogonal matrix [21]. If projectors are used to embed spatial codes, such as markers for optical camera tracking as it will be described in chapter 4, this method can be used to guarantee the undistorted projection of a pattern in situations where projections are carried out from a steep angle. To solve this for the required warping matrix, at least four correspondences between projector pixels and the target perspective have to be known. These can be defined manually, for example, by using the mouse to define the four corners of the target projection, or automatically by using a camera to record at least four individual projector pixels projected one after another or encoded via structured light projection techniques [142]. These 2D to 2D relationships are used to calculate an perspective warping matrix which defines the relationship between all pixels on the projectors' image plane to the defined region of the planar projection surface. Figure 2.3 describes this calibration procedure. If multiple projectors are used, they all have to be registered to the same reference view. Note that this approach does not consider any non-linear distortion effects, such as, for example, optical distortions of the projection and camera lenses and thus may not be able to generate a completely pixel-precise relationship between projector and camera pixels if the latter is used for calibration and no image undistortion algorithms are applied.

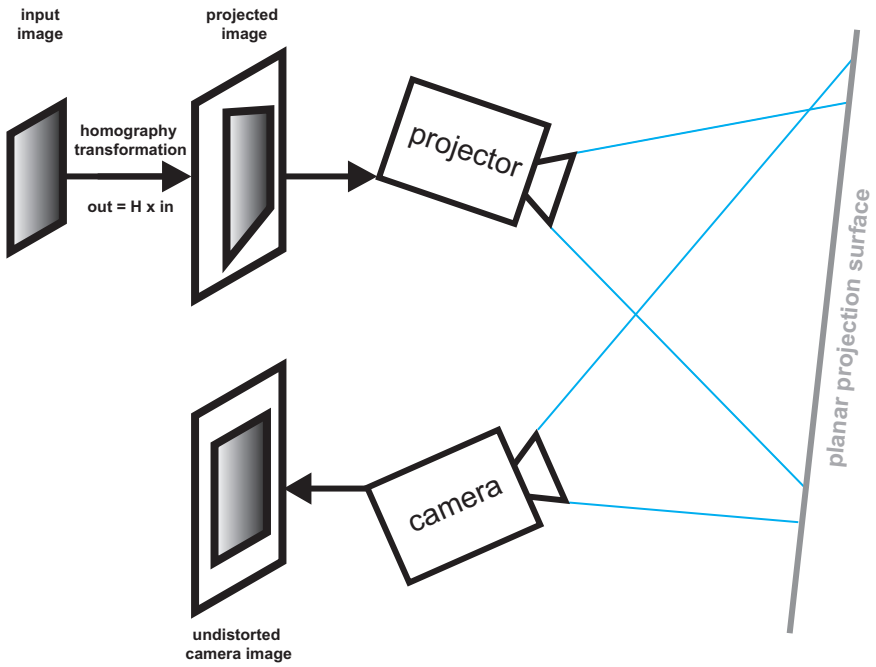


Figure 2.3: Geometrical calibration in the case of planar surfaces using homographies: The input image is warped to appear undistorted from the desired target perspective (in this case the camera's viewpoint) by multiplying its coordinates with the calculated homography matrix.

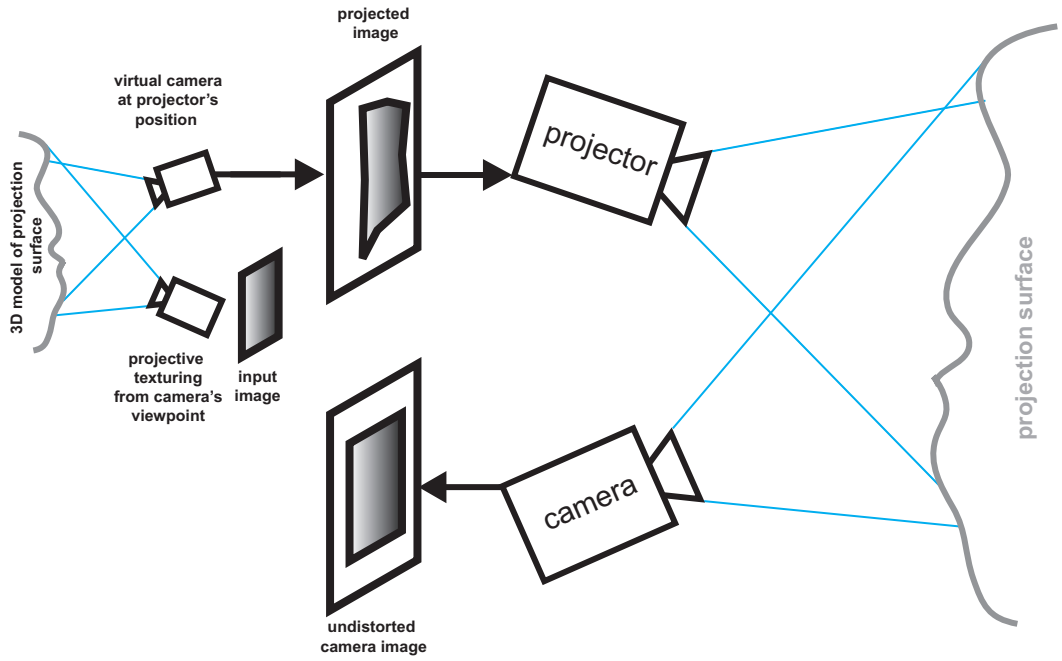


Figure 2.4: Geometrical calibration via projective texture mapping: The calibrated projector and camera data is used to render images with the input image projectively textured onto the geometric model of the surface to make it appear undistorted from the perspective of the (real) camera.

If the 3D geometry of the projection surface is complex, but known, another concept can be used to generate an undistorted projection. For this, the *intrinsic* parameters, i.e. the optical properties, as well as the *extrinsic* parameters of the projector, i.e. the spatial position and orientation with respect to the geometry's coordinate system, have to be calibrated in advance. In that case, projective texture mapping can be used to project images onto the surface in such a way that they appear undistorted from a desired, possibly dynamic, reference viewpoint, which is normally defined by a virtual camera. This is carried out by rendering a texture of the desired image using projective texture mapping onto the geometric model for the defined virtual reference camera. This scene is rendered from a camera's viewpoint with the projector's extrinsic and intrinsic parameters. If this image is then displayed by the corresponding projector, the physical projection appears to be undistorted from the perspective of the reference camera. Figure 2.4 describes this calibration procedure. While this approach makes it possible to project undistorted images onto arbitrarily complex surfaces, it requires precise calibration of the projector's intrinsic and extrinsic parameters (which cannot be measured without the complete absence of registration errors) as well as a high resolution three dimensional model of the projection surface, which makes this method inap-

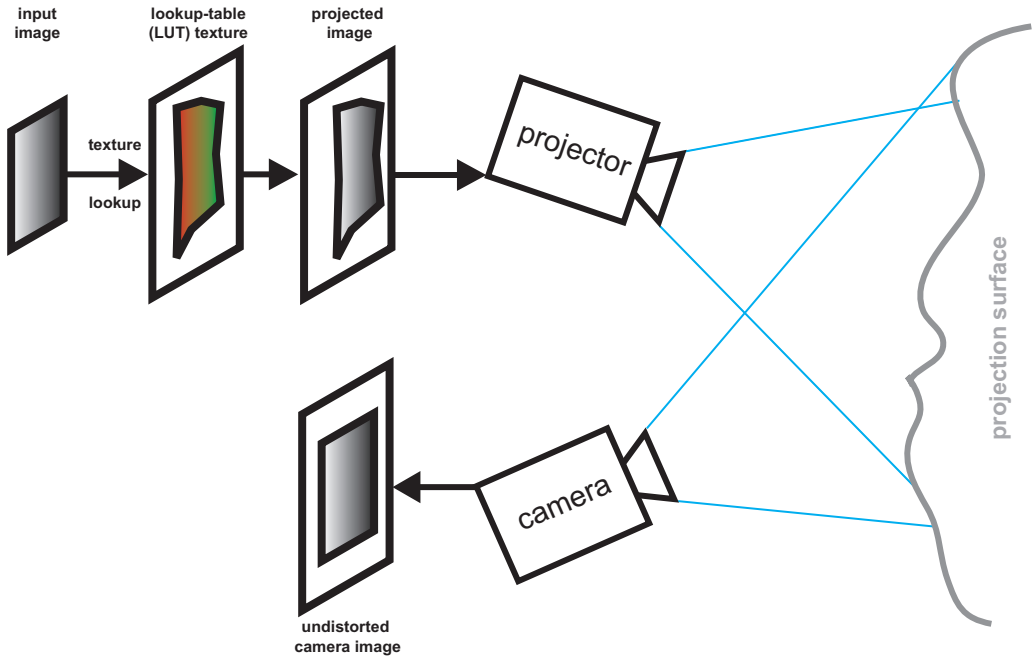


Figure 2.5: Geometrical calibration via per-pixel warping: Each individual pixel of the input image is warped via dependent texture lookups from the desired perspective onto the perspective of the projector via a calculated lookup table to generate an undistorted impression from the camera's view point.

plicable for many situations, such as ad-hoc setups. In the following, a method will be described which overcomes these constraints.

If the geometry of a complex surface is not known, a camera-based calibration technique can be applied to generate a per-pixel lookup table from each projector pixel to its corresponding camera pixel. Because of the impracticability of projecting thousands of pixels one after another to generate the lookup table, acceleration structures are used which encode multiple relationships in parallel. This is carried out by projecting *structured light patterns*, such as Gray Codes [18] or other patterns [142], onto the surface [7]. The structured light patterns encode unique IDs for each individual pixel for example as a series of binary codes in the case of Gray codes. These are captured by the camera which defines the desired target perspective. The recorded images are binarized and further processed to calculate the correspondence between individual camera and projector pixels. In a final step, holes are filled via bilinear interpolation in the generated projector-to-camera lookup table to calculate a dense mapping from each projector pixel to each camera pixel. Figure 2.5 pro-

[7] A comprehensive overview of the different methods and their advantages as well as disadvantages is given in [53]

vides a schematic overview of this registration method.

Depending on the application scenario, each of these methods has its distinct advantages. If, however, the projection surface's reflection properties are not uniformly white and/or textured, a pixel-precise relationship is required which makes the lookup-table based approach most suitable for applications where the surface's reflection properties need to be compensated for. How this can be achieved is explained in the following section.

2.1.3.2 Radiometric and Photometric Compensation

To compensate for the influence of a non-perfectly-white reflecting projection surface, information about its reflectance properties has to be acquired during initial calibration. This can be carried out efficiently by combining this calibration step with the camera based lookup-table approach for geometric registration explained in section 2.1.3.1: in addition to the structured line codes required to calculate the pixel relationships, additional images are projected to acquire the surface's reflectance information for each individual camera pixel. This approach makes it possible to correct the spatially varying reflectance behavior on a per pixel basis by adjusting the projected intensities such that the influence of the colored surface is neutralized.

A series of different radiometric and photometric compensation techniques exist which focus on different fields of applications such as a perception-based, content-dependent globally and locally optimized photometric compensated projection for off-line [4, 170] and on-line systems [67], the use of multiple projectors [13], head-tracked, stereoscopic applications [17], compensated projection onto complex, non-lambertian surfaces [123, 175], and adaptation to dynamic environmental lighting conditions [94] as well as for adapting compensation onto dynamic surfaces in real time [57]. In the following section, we will introduce the basic radiometric compensation techniques for lambertian surfaces which were applied in the context of the work in this thesis. Please note that in the following each identifier presents the RGB values of an individual pixel of a two dimensional image.

When projecting onto a non-perfectly white surface, the projected intensity I is modulated by the surface's reflectance properties M. The amount of incident light depends on the geometric relationship between the light source and the surface point, i.e. the *form-factor* F, and can be approximated by:

$$F = \cos(\alpha)/r^2 \tag{2.2}$$

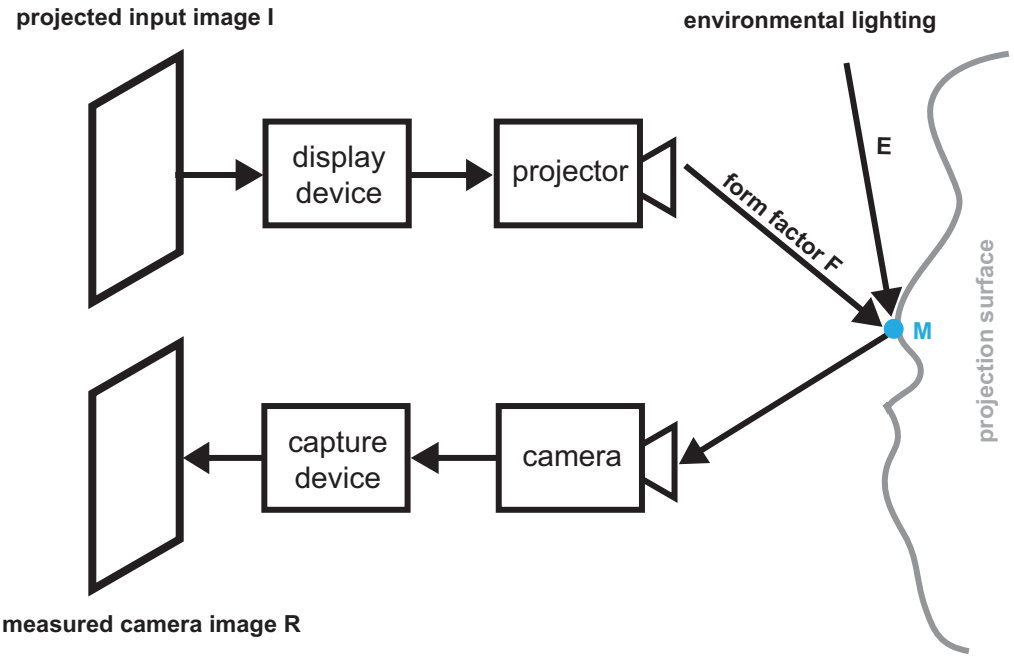


Figure 2.6: Flow diagram describing the individual steps which have to be considered to generate a compensation image neutralizing the contributions of the surface's material properties and environmental lighting. The input image I is modulated by the device's response curve and projected onto the projection surface. It's intensity is attenuated by the form factor F and modulated by the surface point's reflectance properties M. The reflection is recorded by the camera and further processed by the capture device to finally produce the output image R

Where α equals the angle between the light ray and the surface normal and r^2 describes the square distance attenuation of the light's intensity. In addition, existing environmental light E will be added to the reflected intensity. Thus, the recorded camera image contains a mixture of all of this information [8]:

$$R = I\cos(\alpha)/r^2 M + EM \tag{2.3}$$

Where R equals the reflected radiance, which can be recorded by a camera. After substituting the form factor, this relationship can be described as:

$$R = IFM + EM \tag{2.4}$$

Using a camera to record this information makes it complicated to separate the material properties M from the form factor F and the amount of environmental light E. This, however, is not required to

[8] The form factor between camera and surface as well as further camera based intensity modulations such as optical vignetting etc. are not considered within this model

calculate a compensation image. Therefore, the products FM and EM are measured by projecting a fully white image to measure FM and a dark image with zero projector intensities to measure EM, which consists of the uncontrollable amount of environmental illumination as well as the projector's black level. Note that the recorded intensities of EM have to be subtracted from FM, to remove the contribution of environmental illumination contained in the recorded FM. Figure 2.6 summarizes the light path from the projector lamp to the camera sensor: by defining R as the desired input image O, a compensation image can be calculated by:

$$I = \frac{O - EM}{FM} \tag{2.5}$$

If this intensity is projected onto the surface by the geometrically registered projector, the desired image intensities will be recorded by the camera. Note that these linear relationships require a completely linearized setup: the possible non-linear internal processing of the display as well as capture device (cf. figure 2.6) is not considered. This necessitates the initial measurement and correction of the projector's and camera's non-linear response behavior which will be described in chapter 7.3. Besides the necessary linearity, this compensation algorithm does not consider any overlapping of RGB filters used to generate the impression of color. While these filters do not act as perfect band passes and may also not be consistent within the projector and camera used, *color mixing* between these channels occurs, which can lead to slight chrominance shifts in the projected compensation image. This effect can be considered by introducing a *color mixing matrix* into the compensation algorithm [32, 65]. For this, in addition to a white projection, the reflections of the projected intensities of all three color channels have to be measured independently by the recording camera and the resulting data has to be stored in 3x3 color mixing matrices V for each pixel which leads to a total number of nine values that describe the intensity modulation between camera and projector for each pixel:

$$V = \begin{bmatrix} V_{RR} & V_{RG} & V_{RB} \\ V_{GR} & V_{GG} & V_{GB} \\ V_{BR} & V_{BG} & V_{BB} \end{bmatrix} \tag{2.6}$$

The entries of the matrix can be estimated by the captured sample images as described in [65]. By using this matrix, the intensity modulation of the projected images can be described as a simple per-pixel matrix multiplication:

$$R = VI \tag{2.7}$$

Consequently a compensation image I can be generated by multiplying the input image O with the pre-calculated inverse color mixing matrix:

$$I = V^{-1}O \tag{2.8}$$

In the method presented by Yoshida et al. [186], the 3x3 color mixing matrix was extended into a 3x4 matrix which makes it possible to take into account the contribution of environmental light. This method was applied for the system which will be presented in chapter 5.

In case that the projection surface is not perfectly diffuse, a radiometric compensation can be achieved by measuring the full forward light transport between projector and camera, as explained in [145]. The resulting light transport matrix T is much more complex than V, because of the fact that it contains all detectable global and local modulation effects, including interreflection, specular reflection, subsurface scattering, refraction, caustics, and projector defocus. The technique explained in Wetzstein and Bimber [175] can be used for computing the inverse light transport matrix T^{-1}. This allows the compensation of the measured light modulation effects with

$$I = T^{-1}O \tag{2.9}$$

View-dependent modulation effects, such as specular reflections or refraction, however, can only be compensated from the initial calibration perspective of the camera, which implies that the radiometric compensation is only valid for that distinct position.

Independent of the algorithm used, the overall brightness of the projected images has to be reduced to generate a flawless compensation image. Without this, the limited maximum lumen throughput of the projector as well as the amount of environmental illumination leads to clipping errors in extremely bright as well as dark regions in which the surface's material properties cannot be compensated for.

The complete geometrical correction and radiometric compensation can be carried out in real-time by encoding the lookup-table as well as all required color information as texture data and using programmable fragment shaders for dependent texture lookups to efficiently calculate the pixels of the warped compensation image in parallel. This approach was used in this thesis to project spatial codes for projections onto everyday environments which were then used for the generation of digital video composition effects within arbitrary environments (cf. chapter 5). For this, the recorded intensities of the camera image had to be as close as possible to those of the desired code image. Because of the fact that the compensated code images were only required by the camera to generate the compositing effects, it was not necessary to apply high-quality perceptually uniform projection as described, for example, in [4].

2.2 Related Work

Utilizing synchronized illumination modulation for digital video compositing can be achieved in various ways, for example, by directly projecting code patterns for keying onto surfaces, by lighting scenes in special ways to extract information or by hiding coded patterns within the illumination. The systems which have been developed within the context of this thesis are therefore related to a variety of research areas which will be listed below.

In the following sections we will summarize and classify the various related approaches of digital matting techniques and explain the different techniques which were developed to suppress the unwanted effects of color spill and enumerate the differences to our proposed approach. Furthermore, we will report on related projector-camera-based methods which were developed for virtual studio productions, digital video compositing and imperceptible coding methods. Finally we will discuss different solutions for optical marker tracking techniques.

2.2.1 Matting Techniques

The generation of alpha mattes for creating high-quality composites is a field of long and ongoing research. As already mentioned in section 2.1.1, its trivial solution is underconstrained and thus a series of algorithms were developed to optimally estimate the desired result by applying various mathematical methods, constraining the scene's degrees of freedom in terms of colors and camera setup or increasing its quality by using specialized hardware, capturing multiple images in parallel or sequential time order, with different wavelengths or polarization angles. Depending on the application scenario, ranging from real-time TV-broadcasts in a completely controlled studio environment to large-scale outdoor film shots, every method has its distinct benefits, but also drawbacks.

In the following we will summarize the different matting techniques, discuss their advantages as well as disadvantages and classify them depending on their demands and quality. Depending on the hardware setup used as well as the technique to generate the matte, the methods can be subdivided into the following terms:

- Chroma Keying
- Luma Keying
- Infrared (IR) Keying

- Difference Keying
- Flash Keying
- Defocus Matting
- Depth Keying
- Polarization Keying
- Natural Image Matting
- Environment Matting

These are discussed in detail in the following sections.

The predominant method for generating alpha mattes is the *chroma keying* method [166, 6, 9, 28, 45, 108, 30, 8] which requires a uniformly reflecting colored background. Usually, this is realized by homogeneously illuminating background walls which are painted with specialized paints having a highly saturated blue or green color tone [139, 92]. These so-called blue or green screens are used to separate foreground from background pixels depending on the recorded color values and to define transparency values which are stored in the alpha matte. Obviously chroma keying constrains the colors of foreground objects to those that clearly differ from the color of the backdrop to avoid them from being falsely classified as belonging to the background region. Another approach to solving the matting problem in a way similar to standard chroma keying was proposed in [151]: two different backings are used to calculate alpha mattes via triangulation matting. This allows the calculation of correct alpha mattes for arbitrary foreground objects. For this the two different backgrounds also have to be recorded without any foreground objects which means that this approach is not applicable for dynamic video recordings.

If a clear spatial separation of foreground and background objects is available, *luma keying* [23] can also be applied by carefully illuminating only foreground regions and directly applying the luminance signal of the recorded camera image as alpha matte. This method however, can lead to holes in dark or shadowed regions of the foreground object and it is, therefore, important to carefully illuminate the objects.

By using the non-visible electromagnetic spectrum as, for example, wavelengths of the near or far infrared (IR) spectrum, alpha mattes can also be generated by illuminating the background regions with IR-lighting [44, 95] applying the IR illumination to the foreground regions, or directly

measuring the typical thermal radiation of living foreground objects such as humans [183]. While such *infrared keying* methods are imperceptible to the human eye, it requires two optically aligned cameras and care has to be taken to clearly separate the IR wavelengths required for keying and the visible illumination required to illuminate the scene. While this can be achieved by applying appropriate band-elimination filters in front of the light sources as well as the cameras, this method is not appropriate where environmental illumination cannot be controlled and where hot objects emitting IR-radiation may be present.

For scenes recorded with a stationary, static camera system, *difference keying* can be applied by initially recording an image of the scene which contains only background elements (i.e. without any actors or other foreground objects) and generating the alpha matte by calculating the pixel color differences to the current frame. This difference keying method [132] cannot, therefore, be applied to moving cameras or ones with varying optical settings. Furthermore, it is error-prone in non-static lighting conditions, where foreground and background colors are similar or where background objects move.

Using a synchronized illumination makes it possible to generate alpha mattes via *flash keying* by alternately turning either the foreground or the background illumination on and off in every other frame. This makes it possible to generate an alpha matte by calculating the pixel's color differences. While this method overcomes some of the limitations of difference keying, this method also requires careful lighting setups: If the foreground is flashed, shadow casts or dark foreground objects generate holes in the matte. If the background is flashed, background lighting reflecting or scattering onto the foreground as well as dark foreground elements also reduce the quality of the alpha matte. Another flash based approach was presented in [160]: Camera-synchronized blue LEDs are used for chrominance based real-time flash keying. The LEDs are activated in every second frame of a high speed recording for illuminating the foreground with the blue keying color and an alpha matte is pulled from these frames via chroma keying. The remaining frames are normally lit and provide the foreground. Using a flash and a non-flash image pair for matting was proposed in [155]. While this method can handle slight motions correctly during the acquisition of the two required images, it is only able to generate binary alpha mattes which leads to errors if the foreground contains semi-transparent items such as hair or glasses.

defocus matting [107] uses multiple, pixel aligned synchronized video streams with varying focal depths to automatically recover a matte by analyzing the amount of defocus in the video images. This approach requires complex hardware and suffers from the fact that depth discontinuities are

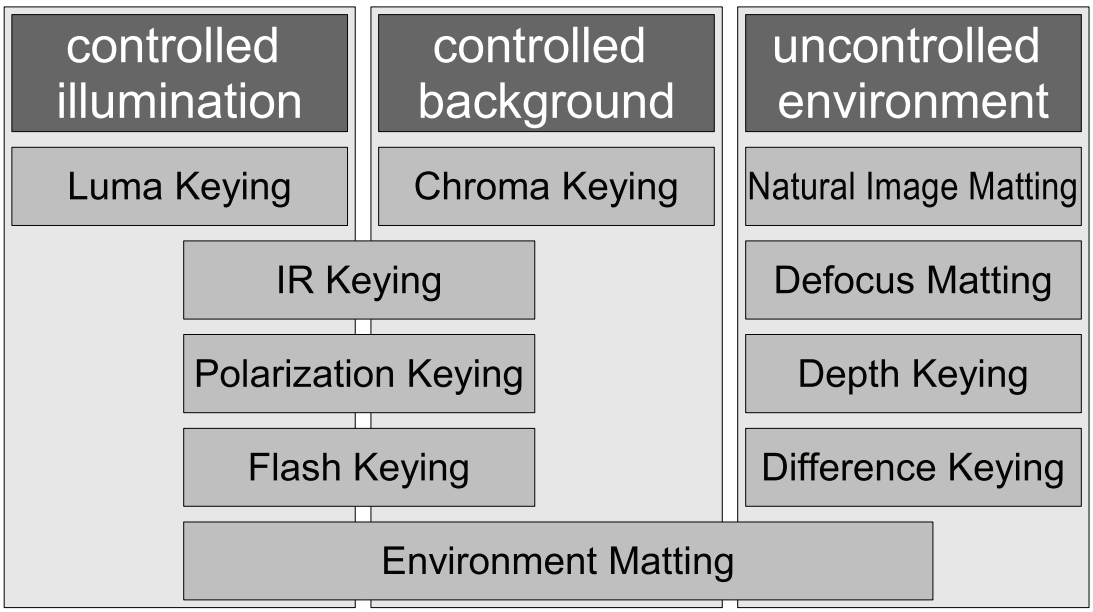


Figure 2.7: Classification of different matting methods depending on the setup requirements.

required to separate foreground from background. In addition, motion blurred or adjacent objects with similar edge colors are misclassified.

A depth-map-based segmentation method is presented in [73]. The depth is estimated in real-time by a time-of-flight (TOF) sensor. Thresholding is applied to the resulting depth map to generate a binary alpha matte. This *depth keying* approach suffers from high-quality non-binary alpha mattes as well as the fact that foreground and background have to be clearly separated. A similar TOF-based approach for mixed reality applications was presented by [90] which, however, suffers from the same limitations in terms of binary alpha mattes and low resolution.

Other methods apply active [10] or passive [106] polarization: two synchronized cameras with differently rotated polarization filters are used in combination with a polarization preserving background. This technique requires a specialized camera or two optically aligned, synchronized cameras to generate the matte. While *polarization keying* is completely color invariant, light reflecting on foreground objects might lead to false alpha values.

A series of semi-automatic methods exist in which a user defined trimap is used to initially define foreground, background as well as unknown regions and different statistical methods are applied to estimate the unknown alpha values: For the unknown regions of the trimap, neighboring foreground and background pixels are analyzed and used to generate statistical information about the

regional appearance of the foreground and background. From this information, the alpha value of the unknown pixel is determined in various ways. In [11] and [12], for example, the surrounding regions are extrapolated into the unknown regions and compared to the actual intensity of the currently analyzed pixel. Depending on this relationship, the alpha value is derived. More complex approaches for high-quality matting are presented in [141, 36, 140, 35, 78, 97, 154]. While these *natural image matting* techniques do not require an artificial, homogeneous background, they tend to fail in situations where neighboring foreground and background pixels have similar colors, as well as in regions without high spatial image frequencies.

Environment matting, which was introduced by [191], is an extension of alpha matting that correctly models reflection and refraction of light from the backdrop plane with the foreground. This allows the correct simulation of complex, indirect lighting effects if transparent or highly reflective objects are used in the composite. In contrast to classical alpha matting, the usage of environment mattes enable a significantly more realistic appearance at the expense of a more complex acquisition process. To gather the required information, most variants of environment matting apply single or multiple backdrop screens and display Gray codes or color ramps in a completely controlled lighting environment [127, 189, 33, 191, 188] to measure the light transport from the backdrop plane to the camera. Since many images must be captured, dynamic foregrounds and uncontrolled camera movements cannot be supported.
A simplification with one image has been described in [37] which makes environment matting applicable for dynamic situations within completely controlled illumination environments, but with this method color spill cannot be handled at all. In [176], an image based environment matting approach is presented which, however, is constrained to certain scenes with a planar background as well as a sufficient amount of textural information. Despite these constraints, the amount of camera and object movement is also limited.

All approaches described so far allow the generation of alpha mattes, but they are all limited in one way or other. While some, for example, require a certain amount of user input and thus cannot be applied in real-time, others fail if foreground and background are similar or if the scene illumination cannot be controlled completely. Table 2.1 lists the different matting techniques and compares their individual requirements, advantages as well as disadvantages. In Figure 2.7 a classification of the described techniques is shown with respect to the basic recording setups required for the individual acquisition method. Because of the fact that some techniques offer different solutions, they can be assigned to multiple categories. While the methods listed in the right column do not re-

Method	Multiple images required	Quality	Constraints	Specialized hardware required
Chroma Keying	no	very good	color dependency/color spill	no
Luma Keying	no	good	shadow casts	no
IR Keying	yes (parallel)	good	controlled lighting required	optically aligned RGB and IR cameras
Difference Keying	yes (temporal)	moderate	static camera pose required	no
Flash Keying	yes (temporal)	good	careful lighting required	synchronized illumination system
Depth Keying	yes (parallel or temporal)	poor	only binary alpha matte possible	optically aligned RGB camera and depth sensor
Polarization Keying	yes (parallel or temporal)	good	reflecting foreground might be wrongly classified	optically aligned cameras or fast exchangeable filters
Defocus Matting	yes (parallel or temporal)	moderate	clear foreground and background separation required	optically aligned cameras or fast exchangeable optics
Natural Image Matting	no	very good	similar neighboring foreground and background colors	no
Environment Matting	depending on applied algorithm	good-very good	no	depending on applied algorithm

Table 2.1: Comparison of the different matting methods.

quire any specialized lighting or background, they require specialized hardware or are constrained in some ways as shown in table 2.1.

Our proposed live compositing approach [70], presented in chapter 4, uses synchronized LEDs for keying. In this system, high-frequency imperceptible flash keying is applied which is combined with an imperceptible pattern projection to support optical camera tracking while generating perspectively correct composites in real-time.
In the system presented in chapter 5 we use projectors in combination with a temporally synchronized bright white LED illumination to pull mattes in front of arbitrary backgrounds by neutralizing its locally varying surface reflectance via radiometric compensation. The spatially coded front projection we use generates a homogeneous luminance on the background surface which makes it possible to pull mattes independently of the foreground and background colors via luma keying. By illuminating the static backdrop surface with one or multiple projectors and turning off the complete foreground illumination, the limitations of luma keying in terms of shadow casts and inhomogeneous illumination can be overcome. Using only the luminance of the projected images to generate the alpha matte allows to embed additional information in the chrominance channels of the projection which can be used, for example, to generate environment mattes, reconstruct the

camera path or to display moderator hints directly on the walls of the film set. While every even frame of the recording is used, the projection is turned off and the LEDs are used for arbitrary scene illumination in every odd frame. In addition, the applied white scene illumination generated by the LEDs can be color-filtered for achieving the desired lighting conditions.
In chapter 6 we propose a temporal chroma keying approach which requires two alternately generated complementary background colors. This approach makes it possible to pull correct alpha mattes independently of the foreground colors without any other user interaction besides initially setting up the chroma keying system used. Most methods based on chroma keying [151],[166] as well as others in which the generated alpha mattes are not perfectly registered to the recorded image [73] suffer from color spill which degrades the quality of the final composite. This is not critical if the chroma keying signal is presented temporally [160], or if the matting process is based on another technique such as natural image matting or polarization keying, however, these techniques suffer from other constraints already mentioned. The next section summarizes existing methods for color spill reduction.

2.2.2 Color Spill Reduction

Today, chroma keying is widely used for digital video composition effects. Its simplicity and robustness has made it the standard alpha matting method for real-time television broadcasts as well as movie productions. While extensive research in recent decades has led to a series of high-quality algorithms [6, 9, 28, 45, 108, 30], which are also partially implemented as real-time software modules or as specialized hardware components, they all suffer from the same limitations in terms of color dependency: as soon as pixels of the recorded foreground objects contain chrominance values similar to that of the backdrop – mostly a pure green or blue color – they will be incorrectly classified as belonging to the backdrop. Thus the actors as well as foreground objects are restricted in terms of their color.

Despite its dependency on foreground colors, however, chroma keying also suffers from the problem of color spill. This term describes the unwanted effect of background color appearing in image regions classified as belonging to the foreground objects. The edges of the foreground objects tend to be mixed partially with light arriving from surrounding background regions (cf. figure 2.8 for an example). To avoid unrealistic appearances this color spill has to be removed in the final composite without excessively changing the appearance of the foreground colors which is a common problem of many color spill suppression algorithms [181].

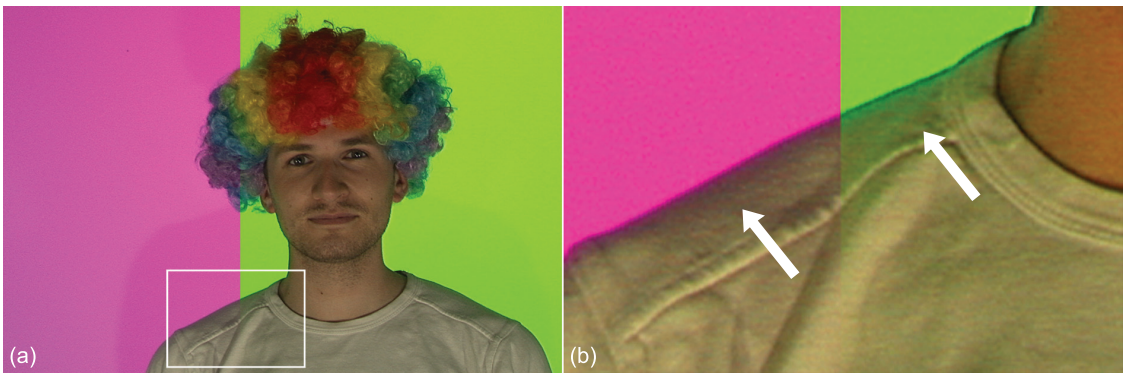


Figure 2.8: An example of visible color spill as a result of background lighting which is reflected onto the subject is shown in (a). For better visibility, the saturation of the zoomed-in excerpt (b) corresponding to the region marked in (a) has been increased. Despite its possible false alpha value classification, the visible color spill on the actors shoulder leads to an implausible composite.

This problem was first addressed in [166], where a solution was presented for blue screen matting by analyzing the maximum RGB color differences at a pixel to classify pixels that are influenced by background spill. This analysis, however, only works correctly for a certain number of color tones such as human skin. This can lead to wrong color reproduction of the foreground in the final composite as this method tends to modify the hue values of certain colors.

Another technique for color spill correction was explained in [151]. The authors propose an idea to solve the color spill problem by calculating an additional spill alpha matte by assuming the color spill is not modulated in its color. However, only the idea was proposed without a proven algorithm, the authors recommend this for future research.

In [73] color spill is removed by utilizing spatial coherence and calculating the correct foreground color by assuming constant foreground and background colors in a 5x5 neighborhood. The foreground color then is extrapolated into the surrounding background regions. This approach leads to problems if the foreground objects contain multiple different material properties.

Another approach requires manual user interaction to classify regions with possible color spill [48] which prevents this method from being used in real-time. By using information about the background color as well as the lighting color, the amount of color spill is estimated by a least-squares method. More advanced spill suppression algorithms were presented in [167] and [29]. Both, however, require the empty background to be captured for each new recording perspective and therefore only support a static camera and fixed lighting conditions.

The reason for the wide application of chroma keying is that, when compared to the alternatives discussed above, it is the most robust technique with the least amount of critical limitations. The dependency on foreground color and color spill are the only two exceptions. In chapter 6 of this thesis a temporal approach is presented which neutralizes color spill automatically and converts it into a neutral background illumination. It is also corrected if objects are in motion during the recording of consecutive frames. Our proposed color spill neutralization method is based on the suggestion made in [151] to use two differently colored backgrounds to overcome the limitations of simple chroma keying. However, it does not require registered images of the two backgrounds. By synchronizing a light generation device with the shutter of the camera, this idea is adapted to video recording and by carefully choosing the two background colors, in addition to the possibility of color independent chroma keying, color spill can also be corrected simultaneously.

2.2.3 Advanced Projector-Camera Systems

Besides multi-projection systems, geometric registration and radiometric compensation, projector-camera systems have been used for other applications such as in the field of digital video compositing as well as for the projection of visually undetectable information. In the following we will discuss types of systems which are related to our work: Firstly we will describe systems in which video projectors have been used in virtual studios to support in-place moderator information, to enhance the keying quality or to simulate shadow casts. Then, we will describe various methods which allow the projections of patterns in such a way that they are not perceived by human observers.

2.2.3.1 Projectors for Digital Video Composition

One of the main disadvantages of virtual studio productions is the fact that actors are not able to directly see the objects which will be superimposed in the final composite. This makes it complicated for them to interact in a natural way and requires careful training [103, 111]. Normally this problem is solved by displaying a real-time composite on a screen visible to the actors located outside of the camera's view frustum. This, however can lead to unnatural behavior by the actors as they have to look at the composite screen instead of the camera.
To address this problem, projectors have been used to display visual hints for moderators on screens that remain invisible in the recorded camera stream. This is realized by mounting an LCD shutter in front of the projectors' lenses which blocks the outgoing light during the exposure time of the

synchronized camera [58]. In chapter 5 [66] we apply a similar method for displaying moderator information except that in our case these are on real (non-optimized) surfaces while supporting camera tracking and keying simultaneously. Shirai et al.[148] propose another approach: they use a combination of chrominance and luminance keying instead of a shuttered projection for displaying moderator information together with diffuse blue or green screens, which also overcomes the problem of presenting in-place moderator information. In the work of Grau et al.[63] a technique is presented for projecting moderator information without degrading the chroma keying quality by using retro-reflective screens as chroma color and projection surface.

A further application example is the work of Wojdala et al.[180], where projectors are used for creating synthetic shadows on real objects cast by virtual objects. In contrast to our approach [68] which will be presented in chapter 5, these techniques rely on the completely controlled environment of a virtual studio and cannot be applied in an arbitrary environment.

In [82] and [44], high-speed projectors are used for structured light illumination in combination with synchronized LEDs and cameras to acquire a 3D geometrical model of a moving person. This is accomplished by using modified hardware to enable synchronization at a frame rate of up to 1500 Hz. Here, slight movements of the actor can be neglected during the projection of the required Gray code patterns. While this systems is able to generate impressive results, it has the disadvantage that it requires expensive specialized hardware components.

A large scale, immersive, distributed CAVE was developed within the Blue-C project [64] in which projectors, LEDs, cameras as well as glass panels containing liquid crystal layers were synchronized with each other. This allows the generation of alpha mattes via flash-based keying by switching the glass panels to transparent mode, activating the LEDs and capturing the image by the synchronized camera. To hide this situation from the user, the active shutter glasses are switched to opaque mode during the camera's acquisition time. While the visual quality of the generated alpha mattes was sufficient for a three dimensional volumetric reconstruction of the actor, they were below the standard required for high-quality digital video compositing.

2.2.3.2 Imperceptible Pattern Projection

The limitations of the human visual system (HVS) make it possible to present a series of still images as moving content. While the HVS offers relatively high spatial acuity, the temporal resolution is comparatively constrained as explained in section 2.1.2. Depending on a series of factors such

as luminance, environmental lighting conditions as well as spatial frequencies and others, humans tend to perceive images as still images, flickering or moving imagery, depending on the presentation frequency. The so-called critical flicker frequency describes the lower boundary above which images will be perceived as animated content. High-frequency temporal modulation of projected images makes it possible to integrate coded patterns that are imperceptible to human eyes due to the limitations of the HVS. These codes can be applied for various applications such as on-line 3D reconstruction or optical camera tracking. Binary codes, for example, can be integrated into so-called *code images* by slightly increasing or decreasing pixel intensities by a certain amount (Δ) depending on the code values, 0 or 1. *Compensation images* are computed that visually neutralize the effects of the embedded codes. If a code image and its compensation image are projected at a high speed, the code is not perceivable. These invisible codes, however, can be reconstructed by recording the individually projected frames with a synchronized camera device. The basics of this principle was first described by Raskar et al. [134] who projected binary images and their complements at high frame rates. This is referred to as *embedded imperceptible pattern projection*.

Extracted code patterns allow, for instance, the simultaneous acquisition of the scenes' depth and texture for 3D video applications [164, 171]. These techniques project colored stripes and their complementary patterns onto non-textured dynamic surfaces to enable a feature based multi-view 3D reconstruction. Therefore the stereo cameras used for reconstruction are synchronized to individual projector frames, while another camera is used to simultaneously recover the surface's reflectance properties by integrating over two consecutive frames.

The temporal dithering of single chip DLP projectors can also be used for imperceptible coding by reverse engineering the high frequency dithering patterns of the projection devices for all input intensities as described in [113]. By capturing an image series of a uniformly white illuminated object by multiple projectors with a synchronized high-speed camera, the individual illumination at each pixel can be reconstructed. Therefore the different internal dithering behaviors of the projection devices are compared to the recorded intensities. This enables applications such as photometric stereo for dynamic scenes or illumination de-multiplexing.

An advanced technique was presented in [41, 40], where a specific time slot of a DLP projection sequence is occupied exclusively for displaying a binary pattern within a single color channel. Multiple color channels were used in [42] to differentiate between several projection units. However, unless the DLP mirror flip sequences within the chosen time slot are not evenly distributed over all possible intensities (which is not the case in practice) this technique can result in non-uniform

fragmentation and a substantial reduction of the tonal values. Since the patterns are encoded independently of visual perception properties, local contrast reductions and flickering effects can be visible in unfavorable situations, such as low intensities and low spatial image frequencies, as well as during the temporal exchange of the code patterns. The possible intensity modification of each pixel's color also leads to slightly miscolored image regions. The authors propose to apply a dithering technique in this case to diffuse visible artifacts .

The approach presented by Park et al. [121] presents a technique for adaptively embedding complementary patterns into projected images. This adaptation is used to adjust the intensity of the embedded code with the goal of minimizing its visibility. It is regionally adapted depending on the spatial variation of neighboring pixels and their color distribution in the YIQ color space. A chrominance analysis is carried out for individual pre-defined pixel blocks. The final code contrast of Δ is then weighted by the estimated local spatial variation. Depending on the values in the chrominance channels (I and Q), the code is exclusively integrated into either the red or the blue color channel. While the basic concept of a dynamic coded adaptation is similar to our method which will be explained in chapter 4, the system described in [121] gives little consideration to the restrictions of human visual perception. Since two manually defined parameters adjust the overall strength of the integrated code pattern, the system is not able to automatically calculate an optimized code contrast Δ. A projection rate of 75 Hz makes it difficult to hide the code during fast eye movements. The problem of an non perceivable exchange of code patterns during runtime is also not considered in this paper.

In [190] an imperceptible calibration technique for radiometric compensation is presented which uses high-frequency temporal modulation to apply geometric calibration and to acquire radiometric information about the projection surface. The techniques described in [190] are based on the fundamental coding scheme which will be described in this thesis. The same principle technique is used in the context of out-of-focus projector blur compensation by imperceptibly estimating the spatially varying point spread functions (PSF) of the projection and deconvolving the input images of the projection [125]. The imperceptible pattern projection was also used in other applications, such as AR based medical surgery [146] and geometrically and radiometrically adaptive projection onto dynamic surfaces [122].

Instead of increasing or decreasing the intensity of a coded pixel by a constant amount (such as in [134]) or by an amount that depends purely on technical parameters (such as the user defined

parameters as in [121], or mirror flip sequences [41]), our method (cf. chapter 4) considers the capabilities and limitations of human visual perception for embedding codes. It estimates the *Just Noticeable Difference (JND)* based on the human contrast sensitivity function and adapts the code contrast (Δ) on the fly, based on regional properties of the projected image and code, such as luminance and spatial frequencies. Thus, only the global image contrast is modified rather than local color values. This ensures imperceptible coding while providing a maximum intensity difference for decoding. Yet, it enforces only a small amount of linear contrast compression. As mentioned above, intensity coding can also be supported with our method, rather than being limited to pure binary patterns, as shown in [190]. Besides the linearization of the projection devices used, no additional calibration is necessary. The integrated codes can be dynamically exchanged arbitrarily during runtime. Because of the fact that an abrupt code transition would be temporarily visible for human observers, this is realized by a temporal code blending technique to seamlessly exchange different code patterns.

Besides these imperceptible projection techniques, other, non-visible wavelengths, such as near infrared lighting, can be used for imperceptible pattern projection [53], which, however, requires specialized hardware: two optically aligned cameras, a spatially modulatable IR illumination device as well as completely controllable illumination conditions. In this thesis we tried to generate the imperceptible codes inside the visible spectrum by directly encoding them into the projected imagery.

2.2.4 Optical Marker Tracking

In chapter 4, we present an adaptive optical camera tracking technique that integrates invisible tracking markers into background projections using our invisible coding approach already mentioned in section 2.2.3.2. Thus, they are displayed directly within the field of view of the camera without being directly perceptible by human observers. In contrast to various marker-less natural feature tracking techniques [55, 159, 39], marker-based approaches do not rely on the robust detection of distinct scene features. If invisible, however, marker-based tracking can share the advantages of a robust marker-less approach in terms of not disturbing the visual impression of the perceived scene. Natural feature tracking methods will fail for uniformly structured surfaces or under dim lighting conditions. This limits their applicability in an arbitrary environment. Marker-based tracking provides artificial visual features by integrating clearly detectable, mostly binary black and white tags. Despite their visibility, occlusions and dynamic reconfigurations of the installations, however,

cause additional problems.

Visibly projected markers have been used previously for geometric projector calibration on planar screens [52]. In chapter 4, we propose a dynamic multi-resolution approach to ensure continuous in-shot camera tracking rather than projector calibration. In contrast to similar nested marker techniques [157] that embed several code scales in a single printed pattern, our multi-resolution markers are projected and can consequently be automatically exchanged and modified depending on the actual camera pose and possible occlusions of the background projection. This makes the approach much more flexible and adaptive for arbitrary, unconstrained camera and object movements.

Another imperceptible marker tracking approach was presented by Park et al.[124]. They use a coaxial camera pair to capture the scene with and without an infrared (IR)-filter. The physical marker codes painted with IR-reflecting ink remain invisible and can be detected when illuminating them with IR light. Our goal is to support camera tracking with dynamically projected markers that are imperceptible for live observers and that are also not visible in the recorded video stream. A key requirement, therefore, is the ability to separate foreground from background. While some other approaches try to overcome this problem using special tracking hardware, others try to estimate the camera's pose by observing natural features or artificial tags [159] with additional cameras (marker-less and marker-based optical tracking).

A very common technique for virtual studios is to integrate markers directly into the blue screens by painting them in a different blue tone that does not affect chroma keying. An example is the widely used ORAD [156] system or other advanced techniques, such as [182]. They support efficient in-shot camera tracking in virtual sets. However, within augmented studios or real film sets markers should neither be directly visible to the audience, nor become apparent in the recorded video stream. Consequently, marker-based tracking within such environments is usually restricted to observing out-shot areas such as the ceiling or the floor which are normally covered by studio equipment, such as light installations, cables, and mountings.

3 Synchronized Illumination Modulation for Digital Video Compositing

In this thesis we present three different hardware systems that use spatially and temporally modulated illumination synchronized with the recording camera to generate various visual effects which can be used to simplify digital video compositing and to open up new application possibilities in this context. As already mentioned in the introduction, the aim is to develop flexible approaches that overcome the limitations of traditional solutions such as the need for virtual studios or the application of standard chroma keying with its associated limitations. In addition to solutions for efficiently generating high-quality alpha mattes, we show how these systems can be used to overcome the limitations of classical chroma keying, to generate environment mattes, to support camera tracking and match-moving algorithms, to display dynamic in-shot moderator information and how these effects can be generated in arbitrary environments.
In the following section we discuss the requirements we defined for the developed systems and briefly summarize the three different approaches presented in the later chapters of this thesis.

While a series of systems already exist for generating high-quality visual effects, they either require a completely controllable environment, such as a virtual studio, or additionally, require expensive specialized hardware equipment, such as, for example, the lightstage systems presented by Debevec et al.[174]. Our main goal focuses on using standard components such as firewire or studio cameras and off-the-shelf video projectors to offer cheap and flexible solutions for visual effects using digital video compositing for everyday use. Depending on the application scenario, the systems presented can be applied in the context of real-time, live compositing situations or for the off-line generation of various effects in the post-production stage. While all the systems presented use the same basic hardware components – which consists mainly of a spatially and temporally modulated lighting system synchronized to the recording camera – they each focus on different application scenarios as summarized below.

In the following three chapters of this thesis we will discuss different methods of synchronized illumination modulation for digital video compositing. The motivation behind each case are stated to explain the reason for the development of these three specific systems.
The first system, which is presented in chapter 4 provides a real-time compositing solution that enables marker-based optical camera tracking as well as the generation of alpha mattes via flash keying. Synchronizing all components at 120 Hz makes it possible to hide all the required coded

information from the human eye. This is achieved by integrating the markers directly into the projected intensities by slightly increasing and decreasing its intensity depending on the encoded marker data. The main focus of the adaptive imperceptible coded projection was to keep the codes invisible even during extreme situations such as saccadic eye movements or code exchange. This was achieved by deriving the coding parameters from a user study and using these values to dynamically adjust the strength of the embedded codes in real-time during run-time. The non-visibility of the embedded codes was confirmed by user evaluation. Furthermore, a real-time prototypical system was implemented that makes it possible to see a perspectively correct augmentation of the video. For this, the projected tracking markers were dynamically adapted in terms of size and position to guarantee their visibility for the current camera's perspective. In addition, occlusion correct composites can be rendered by using the alpha mattes generated via flash keying.

The second system, presented in chapter 5 focuses on the application of temporal coded projections for supporting various visual effects using off-line post-processing. It makes it possible to generate high-quality alpha mattes outside of controlled virtual studio environments in arbitrary film shooting locations by projecting radiometrically compensated code patterns onto diffuse and/or textured background surfaces to neutralize their reflectance properties. In addition, it enables the simulation of refractions of the virtual background in the real components of the composite using environment matting techniques. The projections can also be used to project patterns for off-line camera path reconstruction via match-moving or to display dynamic moderation hints directly on the surfaces. While these techniques are carried out with the synchronized systems by alternately projecting the code patterns on the even frames and illuminating the scene with LED lighting in the odd frames of the recording, the completely illuminated scene, i.e. all foreground and background elements, can be used in the final composite – no physical blue or green screens are required.

Finally, in chapter 6 we present a third system which offers a practical temporal chroma keying solution overcoming its two main limitations – its color dependency and color spill – through the application of high-frequency alternately switching backdrop colors. Choosing two complementary colors as backdrops for chroma keying and combining both individually processed alpha mattes with a maximum operator overcomes the problem of color dependency: foreground regions which are wrongly classified as background because of their similar color will be classified correctly if the complementary color is used as background. Using a blended image of both backdrop states leads to neutral white back illumination instead of the undesired color spill visible in the individual images. In addition, we also present a real-time solution to overcome the temporal registration problems resulting from this sequential approach.

At the end of each chapter we will briefly summarize the presented systems, describe the limitations of each approach and list possible improvements and specific applications for future work.

Although parts of the individual systems could be combined with each other, we decided to separate them into distinct systems due to hardware constraints as well as the different areas of application, e.g. on-line or off-line processing, or the location of the setting inside or outside of a studio environment. Due to the limitations of the available hardware, we decided to generate various effects using off-line post-processing. The majority of the presented algorithms, however, are not constrained to off-line applications and can also be applied on-line (with a one-frame delay) if sufficient hardware resources are available, e.g. low-latency high-speed image processing pipelines as well as parallel processing units.

An in-depth explanation of the hardware components required and their control and synchronization is provided afterwards in chapter 7. At the end of the dissertation in chapter 8, we conclude with a summary of the possibilities of all the systems presented and discuss various more general application scenarios for future work not carried out in the scope of this thesis.

4 ADAPTIVE IMPERCEPTIBLE PATTERN PROJECTION

The various possibilities of temporally and spatially controlled illumination modulation were realized in different hardware setups depending on their actual needs. In this chapter we present a real-time projector-camera-lighting system which enables the live-processing of temporally and spatially controlled illumination which is imperceptible to human observers. A synchronized camera being able to reconstruct the hidden information enables keying, optical camera tracking, as well as the live augmentation of three dimensional content in the composite. We present a novel adaptive imperceptible pattern projection technique that takes into account parameters of human visual perception. A coded image is temporally integrated into the projected image, which is invisible to the human eye but can be reconstructed by a synchronized camera. The embedded code is dynamically adjusted on the fly to guarantee its imperceptibility and to adapt it to the current camera pose. Linked with real-time flash keying, for instance, this enables in-shot optical tracking using a dynamic multi-resolution marker technique. A sample prototype has been realized that demonstrates the application of our method in the context of augmentations in television studios.

Video projectors can modulate images spatially as well as temporally. Due to the limitations of human visual perception, code patterns can be integrated into projected pictorial content that remain invisible to the observers. However, using high frequency temporal image modulation, synchronized cameras are able to reconstruct the code. The code patterns can then be used for a variety of applications, such as projector calibration, camera tracking, 3D scanning, and other applications [9]. An integration via temporal coding can be achieved by projecting each image twice: a first image containing the actual code information (e.g., by varying the image intensities locally by a certain amount (Δ) - depending on the code) and a second that compensates for the missing image information. The invisibility of a temporally embedded code, however, depends on a number of parameters that vary dynamically according to the projected content and the integrated code pattern. Thus, using a constant contrast window of Δ for coding can lead to the code being perceivable depending on the current projection as well as the hardware setup.

In this chapter we present two main contributions: The first one is an *adaptive imperceptible pattern projection technique* based on high-frequency temporal image modulation that overcomes the limitations of existing approaches. The second one is a *dynamic multi-resolution marker tracking method* that - based on our imperceptible pattern projection - ensures a continuous in-shot camera

[9] See [15] for further examples.

tracking despite possible occlusions and individual camera perspective. When combined, these techniques allow one to display arbitrary projected content that is visible to observers in the familiar way. The poses of synchronized cameras, however, can be estimated with the extracted code patterns. We have also combined our methods with *real-time flash keying* using either a high-speed white-light LED illumination or the temporally coded projection itself.
Finally, we present a prototype system that integrates our techniques into the mock-up of a real-time television studio. We demonstrate camera tracking, keying and the composition of computer generated backgrounds with the recorded image of moderators or actors in an arbitrary studio environment. With this, we show the possible applicability of such techniques in real television studios [15] (e.g., hosting a live audience), rather than being limited to virtual sets only. An overview of this system is given in figure 4.1.

The remainder of this chapter is organized as follows: An overview of our adaptive imperceptible coding technique is presented in chapter 4.1. Next, in 4.1.1 and in what follows we will explain the basic concept and provide details on the image analysis, coding, decoding and code blending approaches. Furthermore, we describe the results of an informal user evaluation that validates our results. The example of a TV studio prototype application is presented in chapter 4.2. After an introducing overview, we show how real-time flash keying is used in chapter 4.2.1. In addition chapter 4.2.2 details our adaptive code placement technique. All computation steps are summarized in chapter 4.2.3. Finally, chapter 4.3 concludes our work and points out potential areas for future enhancements.

4.1 Dynamic Δ-Coded Temporal Projection

Based on the initial suggestions of embedded imperceptible patterns using high-frequency temporal modulation [134] we have developed an enhanced method that projects and captures encoded images and their complements at a speed of 120 Hz. However, instead of increasing or decreasing the intensity of a coded pixel by a constant amount of the code contrast Δ, we compute the just noticeable differences (JND) and adapt local Δ-values on the fly, based on the regional image intensity and spatial resolution of the projected image and embedded code. This ensures an imperceptible coding while providing a maximum of intensity differences for decoding to avoid the destruction of the code through camera noise or low camera responses. The advantages of this approach compared to previous methods have been already discussed in the related work in section 2.2.3.2. In

Figure 4.1: A TV studio mock-up with back projection encoding adaptive imperceptible patterns (a), two consecutive images captured by a synchronized camera at 120 Hz (b, c), computed foreground matte from real-time flash keying (d), extracted multi-resolution marker pattern for in-shot camera pose estimation (e), and composite frame with virtual background and 3D augmentation (f).

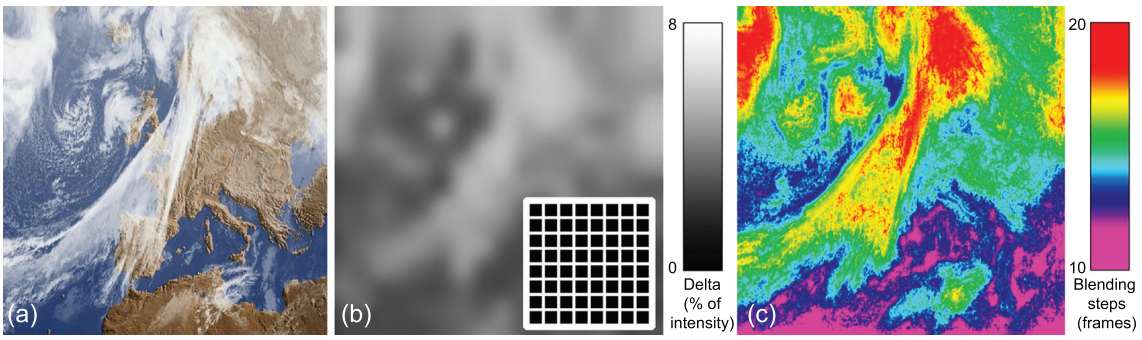


Figure 4.2: For each pixel of the projected image (a) the largest non-perceivable intensity variation is computed (b - contrast enhanced). Together with the code's local spatial frequency (b - down-scaled inlay image) locally optimized Δ-values are derived. From the same information, the smallest number of required blending steps (c) for dynamic code transitions are derived (c - color coded for each pixel)

this chapter we explain how binary codes are imperceptibly embedded into arbitrary images based on parameters of human visual perception.

4.1.1 Static Codes

Where a static binary code image C is embedded into the original display image O we simply compute the projected image with $I=O-\Delta$ and its complement with $I'=O+\Delta$. Projecting both images at a speed faster than the *critical flicker frequency* (cff) [110, 85], a human observer will perceive roughly $(I+I')/2$ which approximates O (cf. background in figure 4.1a). Depending on the binary code bit in C (i.e., 0 or 1) we decide whether Δ is positive or negative on a per-pixel basis.

To avoid clipping at lower and higher intensity levels when subtracting or adding Δ, O has to be scaled. Theoretically a contrast reduction of 2Δ is sufficient. However, for the projector and camera hardware currently used, the brightness of the original image has to be increased by approximately 10% to reduce camera noise in dark regions. Practically speaking, this leads to a maximum contrast reduction of ∼10-20% at the moment. This can be reduced significantly by using cameras and optics that are less sensitive to noise, or brighter projectors. Compared to other approaches, such as [41] (where large tonal shifts for lower intensities in individual color channels or maximum dynamic range reductions of up to 50% are reported), O is linearly scaled in our case. It does not lead to regional color or intensity artefacts.

Synchronizing the camera to the projection enables both images to be captured separately (cf. figures 4.1b,c). Dividing or subtracting them allows identifying the encoded state per camera pixel (cf. figures 4.1e): The ratio of both images are above or below one, while the difference of both images is above or below zero - depending on the integrated bit. It is essential that cameras and projectors are linearized to avoid an effect of their transfer or response functions. A gamma correction can be applied after linearization to ensure color consistency. Thus, projecting and capturing *I* and *I'* at 120 Hz leads to a perception of *O* and to a reconstruction of *C* at a speed of 60 Hz. Despite the integration of binary codes, our approach makes it possible to embed and reconstruct

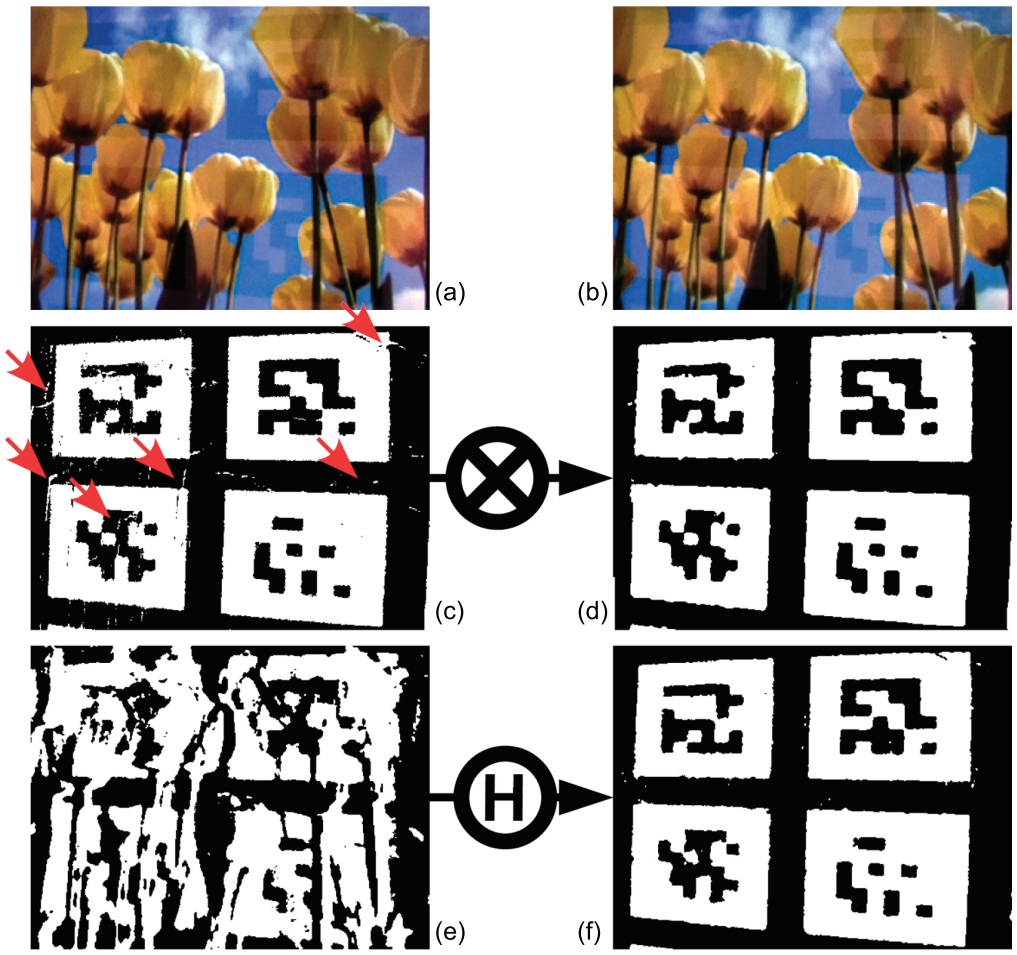


Figure 4.3: Camera movement leads to a misregistered image pair due to the temporal offset of $8.\overline{333}$ ms between code- and compensation image (a,b). The errors in the reconstructed code (c) (red arrows) can be minimized by applying a 5x5 median filter (d). Strong misregistrations that are due to slower capturing rates lead to a complete destruction of the embedded code (e). By calculating the optical flow between the image pair, the image can be re-registered (f).

multiple code intensities at each pixel up to a certain extent. This gives the opportunity to encode more information at the same time. Based on the fundamental techniques described in this thesis, an intensity coding for the imperceptible calibration of projector-camera systems has been shown in [190]. Note, that this is not possible with related approaches, such as [41, 134].
One problem with this temporal coding approach, however, is that for fast camera movements *I* and *I'* may be no longer geometrically aligned. Although the misalignment will not be larger than a few pixels (due to the high capturing rate), reconstructing the code bits on a per-pixel basis might still fail due to the geometric misregistration. By applying a 5x5 median filter kernel, however, these small misregistrations can be removed efficiently (cf. figure 4.3 c,d). In our experiments, this simple technique proved to be applicable for a capturing rate of 120 Hz - even for fast camera movements.
Larger misregistrations that are due to slower capturing rates can be corrected by calculating the optical flow between both images. This is realized by applying a Canny edge detector to both images, compute the optical flow from the result, and filtering outlier flow vectors. With the remaining correspondences, a homography matrix is estimated with which both images can be brought into register (cf. figure 4.3e,f). In this case, we assume that the images are projected onto a planar surface. We found that this technique becomes useful for capturing rates that are below 40 Hz. A more critical problem is that both images also appear misregistered on the human retina during fast eye movements which makes the embedded code become visible. In visual perception research a similar effect is known as the *phantom array* [76, 75] – resulting from saccadic eye movements (cf. figure 4.4 for an illustration). A related effect also appears during the temporal color modulation of DLP projectors, which is better known under the term *color flash effect*, *color rainbow effect* or *color fringing* [22]. Please refer to section 2.1.2.4 for a thorough description of these effects. The strength of the phantom array and consequently the perception of the integrated code during eye movements can be reduced and even eliminated by using only small amounts of code contrasts Δ. If too small, however, the reconstructed code bits are obscured by camera noise. Another solution would be the application of projection devices running at significantly higher frame rates above 500 Hz to make the effect completely disappear [168, 130]. Note that the perception of the phantom array is not a problem seen in related techniques that do not compensate the coded images temporally [41]. For approaches that do perform temporal compensation to avoid contrast artifacts and tonal shifts, such as in our case, this effect can be overcome.

A result of our observations is that the JND of the phantom array and consequently the largest tolerable amount of Δ in a static image depends on several parameters: the regional brightness and

Figure 4.4: The Δ-coded image pair (a+b) is projected alternately which leads to a perceived image (c) for no or slow eye movements (the blue squares illustrate the sequence of projected code image and compensation image). During faster eye movements, however, the code becomes temporarily visible (d). Note that the intensity of Δ is increased in this example for illustration purposes.

spatial frequency of *O*, the spatial frequency of *C*, the temporal frequency of *I* and *I'*, and the speed of the eye movements [22]. Knowing the relation between these parameters enables the dynamic and content dependent regional adaptation of Δ. Since we have not found any literature that reports an exact function which directly correlates these parameters, we have carried out informal user tests to approximate this function. The user tests were carried out in two phases: First a user study was performed with a small number of subjects to estimate the Δ function within a defined parameter range. Then this function was validated (also outside the defined range) during a subsequent user evaluation with a larger number of subjects. Since the speed of eye movements cannot be measured in the normal application case without additional eye tracking hardware, we want to assume fast eye movements as the worst case for the following.

4.1.2 Δ-Function

For estimating the Δ-function, we asked four subjects (one female, three male) to carry out a user study. The participants were questioned to identify Δ at the *JND* point for different projected images with integrated codes. They were sitting at a distance of 95 cm in front of a 110 cm high and wide back projection screen - covering the entire foveal field of view of 60 degrees. The displayed images contained regular checkerboards representing a two dimensional box function with spatial frequencies (*Fb*) ranging from $1/120$ to $1/8$ cycles per degree (cpd) of visual field in both directions, and intensities (*Lb*) ranging from ∼4-27 candela per square meter (cd/m^2). The embedded codes were also represented by a box function with spatial frequencies (*Fm*) ranging from $1/32$ to $1/2$ cpd. Code patterns and image patterns were always phase shifted to avoid cancellation.

Due to the fact that the human eye is most sensitive to green light, the Δ-value of the embedded code was lowered to $1/4$ of its intensity in the green channel - thus the perceptibility of the embedded code could be lowered while the code reconstruction did not degrade significantly.

To guarantee equal conditions, the subjects were given time to adapt to different luminance levels first. Then they were asked to follow a target on the screen that moved up and down quickly at a constant speed to enforce fast eye movements. While changing Δ, the subjects were asked to indicate the point at which the code could just not be perceived anymore (i.e., the *JND* point). This process was repeated about eighty times per subject to cover a combination of five different image frequencies over five luminance levels, and four different code frequencies. The study took about 4-5 hours for each subject - limiting the total number of subjects. The results of all four subjects were averaged and are presented in figure 4.5a. They were later validated during a user evaluation with 28 participants (see chapter 4.1.5).

Due to their mainly linear behavior, the sample points were fitted to planes using multidimensional linear regression (figure 4.5b). The four parameters of each plane shown in figure 4.5b are plotted as small circles in figure 4.5c.

Applying the general plane equation

$$\Delta = -(aLb + bFm + d)/c \tag{4.1}$$

for parameterizing the fitted functions in figure 4.5b requires one to find continuous functions that approximate the discrete plane parameters (*a,b,c,d*) over all image frequencies *Fb*. Figure 4.5c

illustrates the result of a one-dimensional curve fitting:

$$\begin{aligned} \mathrm{a} &= 0.6108\mathrm{Fb} + 0.0139 \\ \mathrm{b} &= 1.144/\cosh(65(\mathrm{Fb} - 0.031)) - 2 \\ \mathrm{c} &= -1 \\ \mathrm{d} &= -0.73914/(1 + \exp((\mathrm{Fb} - 0.04954)/0.01)) + 1 \end{aligned} \tag{4.2}$$

While the parameters *a* and *b* correspond to the gradients of the planes in directions *Lb* and *Fm*, *d* and *c* represent a shift and a scaling of Δ. The scalar *c*=-1 is relative to our study with a temporal frequency of 120 Hz. For other temporal frequencies, it has to be adapted (increased for higher frequencies, and decreased for lower frequencies).

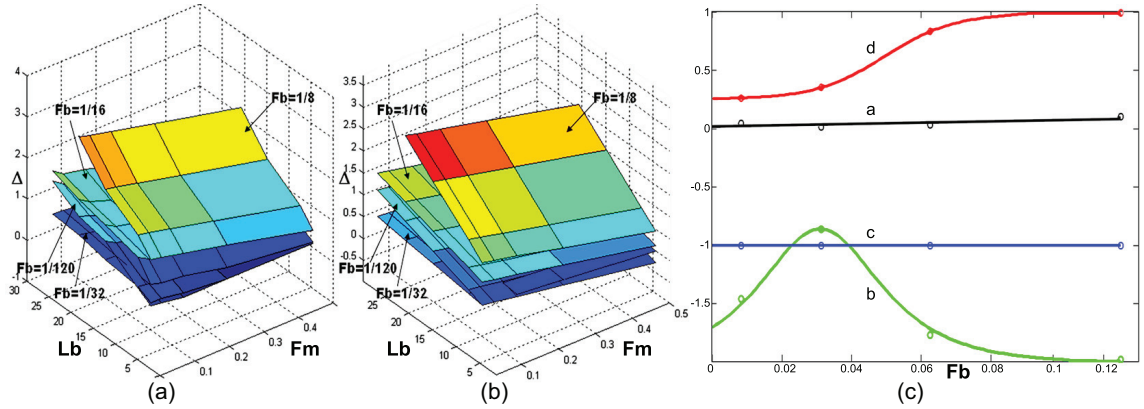


Figure 4.5: (a): Average Δ responses at the *JND* point for a combination of four subjects, four discrete image frequencies (*Fb*), five luminance levels (*Lb*), and five code frequencies (*Fm*). (b): Plane function fitted to sample points (b) for each considered *Fb*. (c): Approximated discrete plane parameters and fitted continuous functions.

Note that the basis functions were selected to fit the samples with a minimal error. We chose a straight line to fit *a*, a trigonometric function to approximate *b*, and an exponential function to represent *d*. With this, the average deviation of our analytical solution with respect to the experimentally acquired values is 0.266 cd/m^2 which equals 0.89% of the projected intensity levels, or ∼2 projected gray scales.

In addition to comparing the analytical solution with the results of the user study, it was also corroborated by values outside our discrete test samples. It was confirmed by the subjects participating in a subsequent user evaluation that the function approaches the *JND* point in these cases as well. Section 4.1.5 summarizes this.

4.1.3 Computation of Δ

Regionally adapting the Δ-values for arbitrary animated or interactive content using the function derived in section 4.1.2 requires the real-time analysis of *O* and *C*.

For acquiring the spatial frequencies of particular image regions, we apply the Laplace-pyramid approach presented in [24]. In our evaluated application cases we found six levels of the Laplacian pyramid to be sufficient for the analysis. As described in [133] we use the absolute differences of each level of the Gaussian pyramid and normalize each of the resulting Laplacian pyramid levels. The results are the ratios of spatial frequencies within each of the generated frequency bands. This is converted to units of cpd, that depend on the observers' distance to the image plane and the physical size of the projection. The input image is transformed into its physical luminance representation in cd/m^2 (the response function of the projector has been linearized and its luminous intensity has been measured with a photometer [10]). With these parameters, we can apply our Δ-function to compute the largest non-visible Δ-value for an arbitrary region within *O* and *C* (cf. figure 4.2b).

As mentioned in section 4.1.2 the visibility of the encoded patterns can be significantly decreased by reducing Δ in the green channel. Decreasing the value of Δ in the green channel down to a fourth of the intensities in the red and the blue channels did not lead to a noticeable quality reduction of the extracted patterns when the maximal difference of all three color channels was used for decoding. Note, that this does not result in a tonal shift of *O* since the embedded code (no matter how large Δ in different color channels is) is always compensated. In practice, for our setup, Δ-values ranging from 0.29 to 1.45 cd/m^2 (i.e., 1-5% of the projected intensity levels, or ∼2.5-13 projected gray scales) were computed.

4.1.4 Temporal Code Blending

Besides the perceived phantom array that is caused by fast eye movements, another visual effect can be observed that leads to the perception of the code patterns in cases when they are temporally exchanged. This is illustrated in figure 4.6. For photopic vision, it can be assumed that the integration times of the human eyes are between 16 ms and 50 ms, depending on the perceived brightness (shorter for bright situations) [110]. If the projected image and its compensation contain a static code over a period of time, the subtraction or addition of Δ at each pixel of both images *I* and *I'*

[10] cf. chapter 7.3

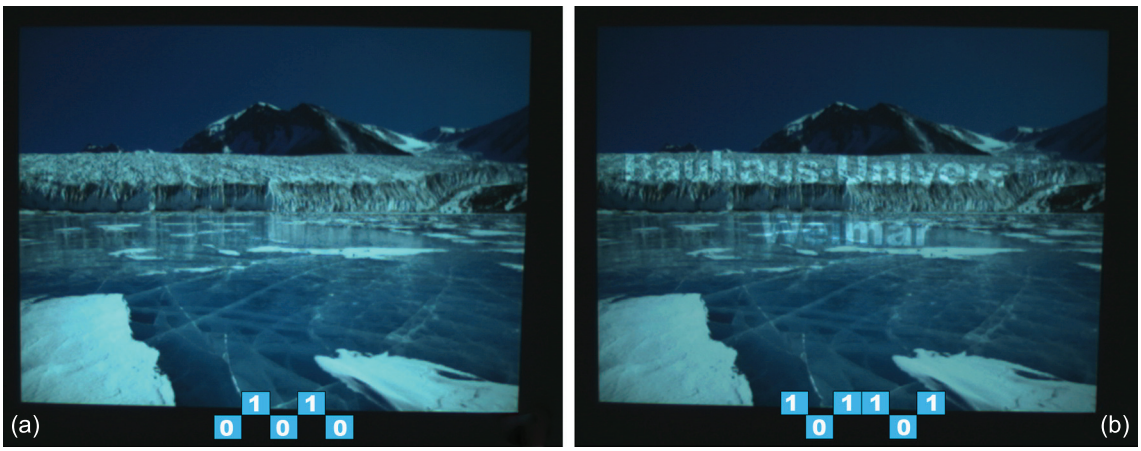


Figure 4.6: If the code is not exchanged (a) it remains invisible to the observer. If an abrupt transition occurs, the correct sequence of code image and compensation is not maintained (b) which results in a detectable flickering.

does not change. Figure 4.7a visualizes this situation. Plotting the relative amount of integrated light for all possible integration times between 16 ms and 50 ms, and for all possible phase shifts [11] leads to the green area shown. The average integration amount, illustrated by a dotted line, is zero in figure 4.6a (assuming no eye movements). Exchanging the code at a particular point in time (i.e., switching from binary 0 to binary 1) leads to the integration results shown in figure 4.6b. The average integration amount during code switching equals Δ, which leads to a visible flickering during this time.

To overcome flickering caused by code transitions, we do not switch between code states abruptly, but temporally transfer from one stage to another stage over multiple blending steps. As illustrated in figure 4.8, the average integration amount reduces to $\Delta/2$ for three blending steps. In general we can say that it reduces to Δ/s for s+1 blending steps if we continuously decrease Δ by Δ/s in each step until Δ=0, and then increase Δ by Δ/s until the original amount is reached. During the center stage (i.e., when Δ=0 and I=I'=O) the code switched.

The maximal average integration amount Δ/s that cannot be detected, and consequently the number of required blending steps, depends on the just noticeable luminance and contrast difference which can be derived from the *threshold-vs-intensity (TVI) function* and the *contrast sensitivity function* as explained in [100, 126]. They are functions depending on the local spatial image frequency and luminance level. Consequently, the optimal number of blending steps for a particular region in O

[11] in contrast to the camera, the integration process of the eyes is of course not in synchronization with the projection

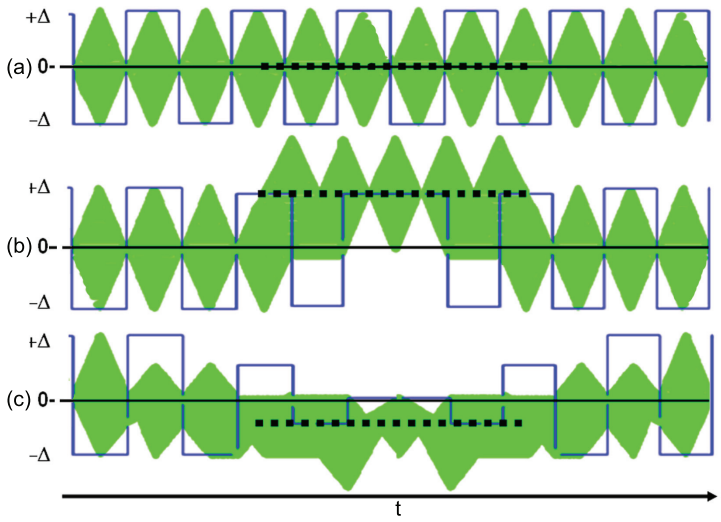


Figure 4.7: This diagram presents the intensity deviations which are perceived during the presentation of imperceptible code patterns. The blue lines illustrate the intensities of the projected code and compensation images. The green area visualizes all possible amounts of perceived intensities that are due to the temporal light integration of the human eye that can vary between and 16 ms and 50 ms for photopic vision [110]. The dotted line represents the perceived average intensity. While in (a) the embedded code is not exchanged, it remains completely unperceivable. During a code transition (b) the embedded code becomes shortly visible during its replacement. If the codes are temporally blended (c) it also remains invisible during transitions.

can be computed from O's local spatial frequency and luminance level by using these functions.

We use the average spatial frequencies and luminance levels of image regions that are already computed for the estimation of Δ (see section 4.1.3). The TVI function and the contrast sensitivity function are applied for these areas and their results are used to compute the *threshold map* as described in [133] for computing the largest not-detectable luminance difference Δ/s (cf. figure 4.8). This leads to the smallest number of individually required blending steps s for each particular code region. If the content in O changes during a blending sequence [12], then the values of Δ and s are adapted and the blending is continued until Δ first decreases to a value ≤ 0, for switching the code, and then increases again until it reaches the new original Δ-value. Varying Δ only by the maximum non-perceivable luminance difference ensures that the code cannot be detected during blending. In practice, 10-20 blending steps were derived [13](cf. figure 4.2c).

[12] e.g., in the case of videos or interactive content

[13] i.e., 3-6 code transitions per second can be supported at 120 Hz

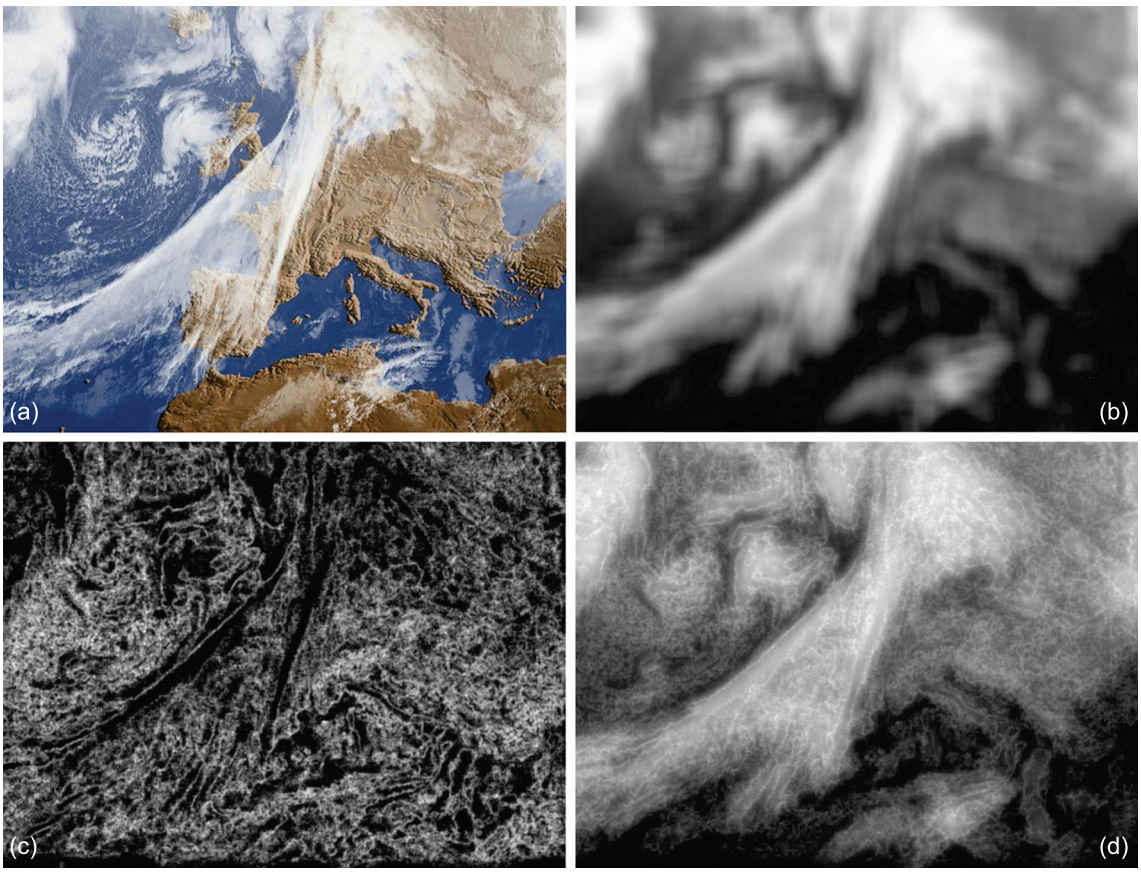


Figure 4.8: Temporal code blending is realized by calculating the TVI values (b) and the contrast sensitivity information (c) from the input image (a) as described in [133] . This information is used to generate the threshold map (d) that is used to derive the number of blending steps for code transitions (b, c and d images are contrast enhanced). In our case the average values for each marker region are used.

4.1.5 Validation of the Δ-Function

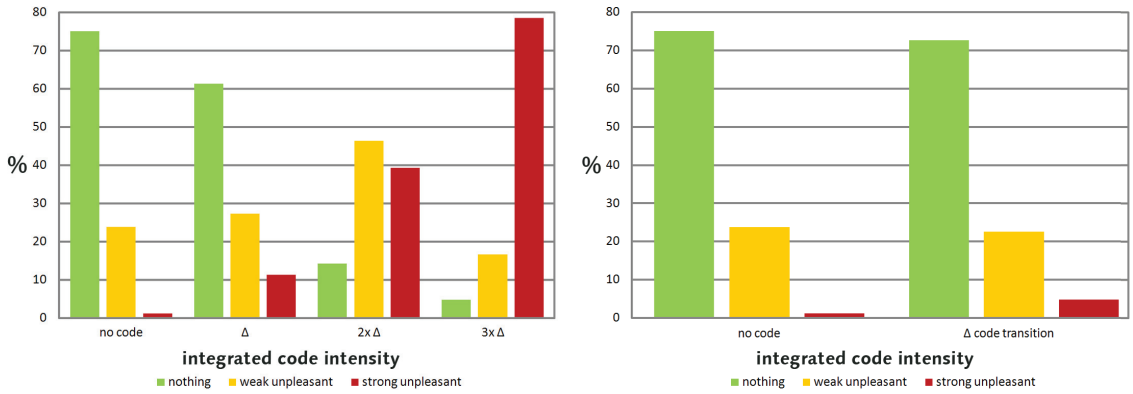
To validate our experimentally derived function for the Δ-coding, an additional user evaluation was carried out. Test subjects had to observe projected images and videos and judge if they perceive any irritating effects in the projection. Note that they were not informed about the integrated code patterns. They were asked to rate their impression in three scales:

- *"nothing unpleasant in the projection"*
- *"weak unpleasant impression"*

and

- *"strong unpleasant impression."*

Each image was projected seven times with varying code and different settings for Δ. While some images did contain an embedded code with the locally derived Δ-intensities, others did not contain any code pattern at all or code with increased Δ-values. Each participant had to rate a series of projections with varying image content and Δ-intensities that were presented in a random order. Furthermore, temporal code blending was applied. Twenty-eight test subjects [14] participated in the evaluation [15] . Note that the image and video content presented contained parameters inside, but are also well outside the initially defined parameter ranges (cf. section 4.1.2).



(a) (b)

Figure 4.9: (a): Average results for delta-coding tests under different scalings. The percentage of each given answer is plotted on the y-axis. (b): Average results of the temporal code blending experiments with and without continuously changing codes.

As shown in the diagram in figure 4.9(a), the participants did not detect any substantial difference between the Δ-coded projection and the projection without any integrated code. But the perception of the integrated code patterns increased rapidly for overcontrolled Δ-values. Overcontrolling was enforced by scaling the Δ-values manually to exceed the computed optimal values. In addition, it could be shown that the temporal code blending was also completely undetectable to the subjects. As illustrated in figure 4.9(b), the participants did not perceive any difference between a projection without integrated codes and a projection that contained code patterns which were continuously blended. As for the Δ-values, the local blending steps were individually computed based on the presented content.

[14] 8 female/20 male, 20-29 years old

[15] The questionnaire can be found in appendix A.4

Another interesting effect that was observed during our user evaluation is the fact that the presentation of animated content leads to a weaker perception of the embedded code, compared to embedding the same code into a still image with the same delta and blending parameters. This appears to be a result of the observers attention to the animation in the presented video. Additional studies have to be carried out to investigate these effects with the goal of further increasing the Δ-values for animated content. If this is understood, the presented video stream could also be analyzed for regions of visual attention by applying such algorithms as proposed in [80, 72, 71] and [184]. This, however, has not yet been done and belongs to our future work.

In addition to our informal user evaluation, our prototype (cf. section 4.2) was presented at a public trade show [16] for the duration of one week. It was viewed by more than one thousand visitors. During the demonstration, none of the visitors questioned perceived the embedded codes.

In the next chapters we present an application example of our coding technique that applies an adaptive code placement for marker-based camera tracking.

4.1.6 Adaptive Code Placement

As explained earlier, for supporting optical in-shot camera tracking, imperceptible two-dimensional markers of different scales are embedded into the projected images (cf. figure 4.1e). In this process the Δ-values and the number of blending steps are computed individually on the fly for each single marker region by averaging the corresponding image luminance and spatial frequency of the underlying area in O and the spatial frequency of the corresponding area in C. For spatial frequencies, the values located within the marker regions of each of the calculated six frequency bands are averaged. The peak frequency is then approximated by choosing the frequency band containing the largest average value.

For a proof-of-concept of the algorithm described, we have realized a mock-up of a TV studio, in which a back-projection screen is used to display dynamic video content with integrated imperceptible codes the in form of two-dimensional markers for optical camera tracking. A synchronized LED-based illumination system is used to lighten the foreground with high frequency flashes. This prototype enables the composition of real and virtual content with respect to the current camera position, and enables real-time keying with the aid of the synchronized illumination. Section 4.2 gives a detailed overview over the individual components of our prototype.

[16] Cebit 2007

To ensure a continuous tracking despite possible occlusions or individual camera perspectives, the code image C is dynamically re-generated, and marker positions as well as their sizes are adapted depending on the current position of the used camera. Consequently, the foreground objects have to be keyed and geometrically related to the projected image for determining occlusions with the foreground. These techniques are described in sections 4.2.1 and 4.2.2.

4.2 Application in a Camera-Tracked Real-Time Compositing System

All of the techniques presented are combined in a sample prototype of an television studio mock-up that is illustrated in figure 4.10: with a moderation desk and a rear-projection screen as backdrop. An off-the-shelf stereo-enabled DLP projector (InFocus DepthQ [46]) displays an arbitrary and dynamic background at a speed of 120 Hz. Our camera system consists of two optically aligned CCD cameras (a). For real-time flash keying, a dimmable 4800 Lumen LED illumination system (c) has been built.

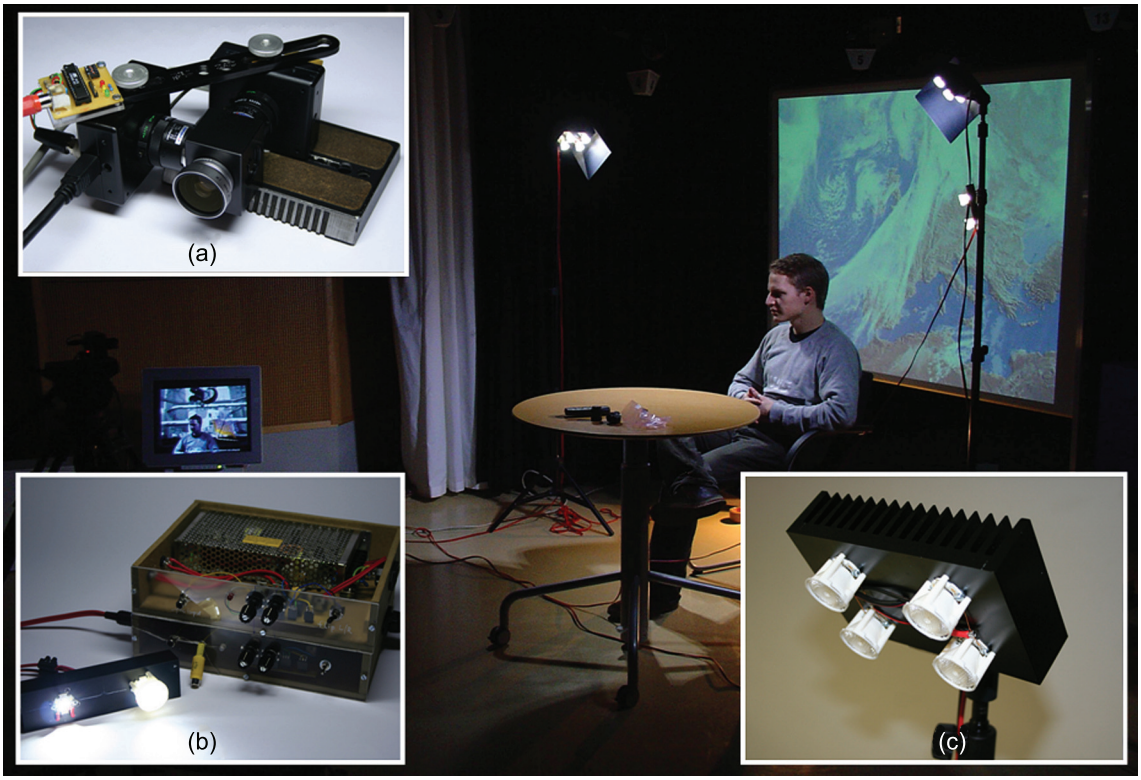


Figure 4.10: Overview over the components of the TV studio mock-up: moderation desk with rear-projection screen and LED illumination, coaxial camera system (a), synchronization units (b) and LED illumination module (c).

Customized synchronization electronics (b) receives the shutter signal that is generated by the graphics card that triggers the stereo projector. This signal is also used for triggering the camera and the illumination system at 120 Hz. The LED illumination can be switched to a flash mode, i.e., on-off sequences, or to a demodulated, i.e., rectified, constant lighting. Point Grey Dragonfly Express cameras deliver raw-format images in VGA resolution over Firewire 800. The prototype runs on a single PC [17]. Please refer to chapter 7 for an in-depth description of the hardware setup.

4.2.1 Real-Time Flash Keying

To separate foreground (e.g., the moderator in the context of our TV studio application example) from background (the Δ-coded background projection), we apply *real-time flash keying*. This is necessary for the image composition and for the adaptive code placement.

By using high performance LED flash illumination we are able to lighten the scene 60 times per second by short flashes with a length of $= 8.\overline{333}$ ms. Thus, every other captured frame contains an illuminated foreground (cf. figure 4.1b), while the remaining frames contain an un-illuminated foreground (cf. figure 4.1c). This allows both to be separated using a variation of flash keying (cf. figure 4.1d). Due to their high frequency, the flashes are not detectable. In contrast to [160] we use white-light LEDs with a white point of 5600K for direct illumination, rather than applying blue LEDs for chrominance masking. Color filters can be used in addition for adapting to the required illumination situation, if necessary.

The matting process in this case is straightforward: Due to the fact that one of the images captured is under full illumination and the other one under no illumination, we can easily extract the pixels belonging to the foreground by analyzing the intensity difference between corresponding camera pixels and comparing it with a predefined threshold. However, care has to be taken because the delta coded projection in the background also differs in its intensity. The difference threshold has to be set to a value that is larger than twice the largest encoded Δ-value. To increase the robustness, we evaluate the maximum difference in the three color channels for thresholding instead of using the average intensity difference of the gray values.

During this process, individual pixels at the transitions of marker boundaries or at the edge of the projection might be misclassified after fast camera movements. The reason is that pixels values in both images might incorrectly contain differences above the defined threshold. As a result, the

[17] Intel Core2Duo 6300, 1.8 GHz / Nvidia Quadro FX 1500 / 2GB RAM

generated matte contains occasional misclassified pixel values. These defects can be removed efficiently by applying a median filter to the generated matte image. In a final step the matte is smoothened by a 5x5 Gaussian filter kernel to soften the transitions between foreground and background. Of course, more advance keying techniques can be applied to generate an optimized alpha matte.

Instead of applying an LED illumination, video projectors themselves can be used to support flash keying if installed in the recording environment. In contrast to simple LEDs, projector-based illumination [14, 135, 93] supports generating a synthetic, spatially varying illumination of real objects on the fly. Thus, in addition to a temporal illumination coding, a virtual lighting situation can be defined, computed and physically approximated within the environment - without changing the physical light sources. Although we have realized flash-keying with projectors using a uniform illumination in our prototype, a combination with a true projector-based illumination technique is a further aspect for future work.

No matter if projectors or LEDs are applied for illumination, flash keying is supported at a capturing speed of 60 Hz for both images. One camera is normally sufficient for this. However, if the Δ-coded projection is out of focus [18] marker tracking might fail. As mentioned earlier, two coaxially aligned cameras (cf. figure 4.10a) are used to avoid this problem: While one camera is focused on the background, the other camera is focused on the foreground. Registering both camera images and synchronizing the capturing process supports recording the focused foreground while processing the focused background. Furthermore, this would make it possible to evaluate relative defocus values of corresponding pixels in both images to enable a depth-of-field based keying, as in [73] and [136]. A real-time keying from defocus, however, has not been implemented yet.

4.2.2 Dynamic Multi-Resolution Markers

As mentioned before, we embed binary markers in our example for optical camera tracking. The stability of the optical tracking strongly depends on a constant visibility of a certain amount of markers with optimal sizes. Moving the camera further away from the screen requires larger markers to avoid a lower tracking quality due to the limited camera resolution. If the camera moves very close to the screen on the other hand, smaller markers are needed to ensure they are visible in their entirety in the camera image. While tracking is not possible if the entire projection is occluded

[18] e.g., due to a short focal depth when focusing on the foreground

from the camera's point of view, an adaptive marker placement leads to a more robust tracking compared to static markers in the case of partial occlusions.

Hence we adjust the projected imperceptible markers within C in each frame by analyzing the visibility of the displayed pixels from the camera's perspective. To ensure the invisibility of the embedded markers during code transitions we apply the temporal blending techniques described in section 4.1.4.

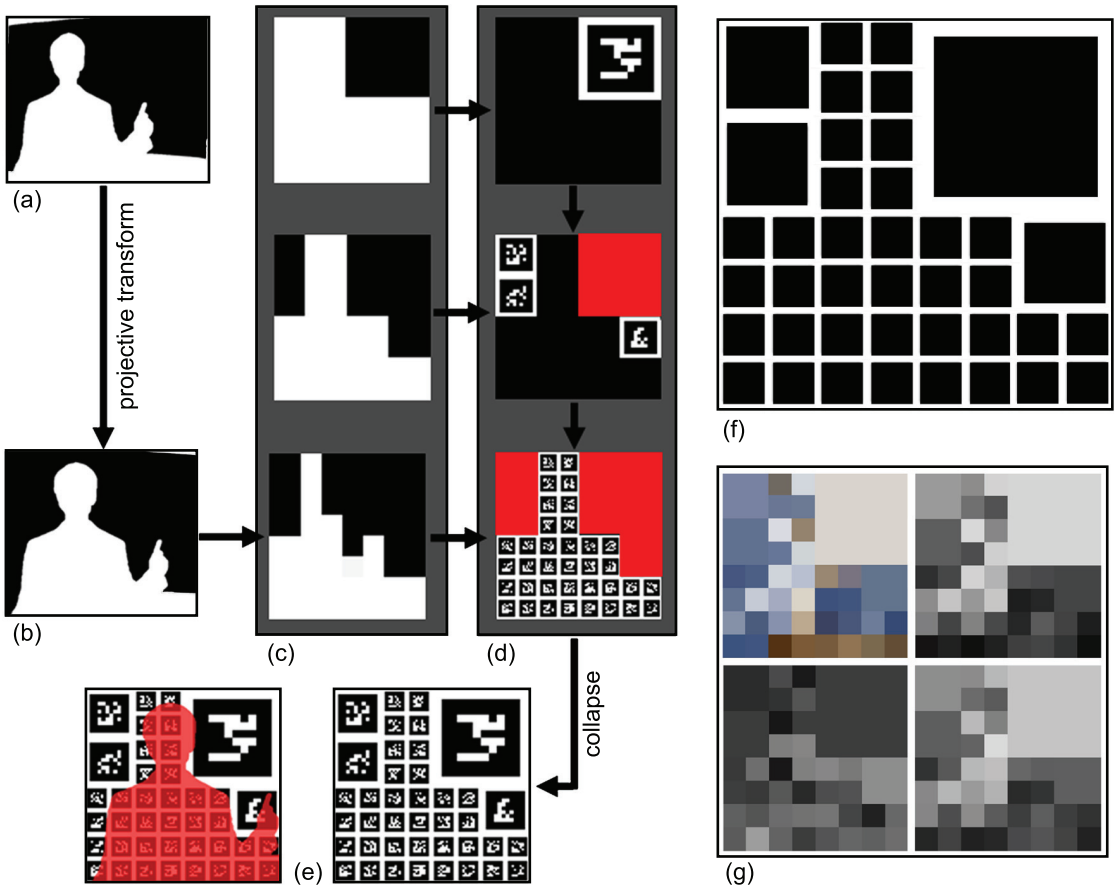


Figure 4.11: Adaptive generation of marker sets: projective transform of foreground matte from the tracked camera perspective (a) to the projection screen (b), construction of visibility tree (c) and labeling of marker tree (d), collapsing of the labeled marker tree (e). The temporal code blending parameters (g, compare with figure 4.8) are based on the average image intensities and frequencies within each marker region (f).

For optical tracking the ARTag library [51] is used which offers the possibility to generate arbitrary array sets from 1024 predefined markers. This feature is used to define a multi-resolution marker

array containing different sized markers for the same spatial locations – all sharing the same coordinate system. We pre-compute a quad-tree that contains multiple markers at different scales in each level. From a higher to the next lower level, the number of markers doubles in each direction while their size decreases by the factor 2. We refer to this as the *marker tree*. Adaptive marker placement is implemented in several steps as shown in figure 4.11.

First, a full screen quad is rendered in projector resolution and a projective transform is computed that maps the generated foreground matte from the perspective of the camera (a) onto it. This is achieved by using the current model-view matrix that results from tracking of the previously displayed frame. The result is a binary image containing the visibility of each projector pixel from the camera's view, which we want to refer to as *visibility map* (b). The principles of this technique are similar to the projective texturing operation required for conventional shadow mapping [179]. Another related method was used in the context of shadow removal within multi projector systems [27].

The initial visibility map is then used to analyze the smallest possible marker size that will be used by geometrically determining the number of projector pixels which are visible in the camera image from the previous perspective. We sub-sample the visibility map into an image pyramid that covers the largest possible marker size in the highest level (e.g., by definition 2x2 markers in C) down to the determined smallest possible marker size in the lowest level (e.g., 16x16 pixels per marker in our case). This leads to a multi-resolution visibility map that we call *visibility tree* (c).

During runtime, the marker tree and the visibility tree are combined at corresponding levels (d): In a top-down direction, only entries that are neither occluded (i.e., marked as visible in the same visibility tree level) nor already occupied by markers of higher levels are processed. The remaining entries are then labeled as occupied within the current level of the marker tree. Regions which are not visible throughout all levels are labeled at the bottom level of the marker pyramid. If the bottom is reached, the labeled marker tree is collapsed and the non-overlapping entries that are occupied by different levels are combined. This results in a code image C that contains the set of optimally scaled and placed markers with respect to foreground occlusions and camera perspective (e). The same constellation from the perspective of the camera is illustrated in figure 4.1e and another example of the adaptive marker placement is given in figure 4.12. As explained in section 4.1.4, local marker regions have to be temporally blended if a code transition within a particular area in C occurs to avoid the visibility of the embedded code within this time.

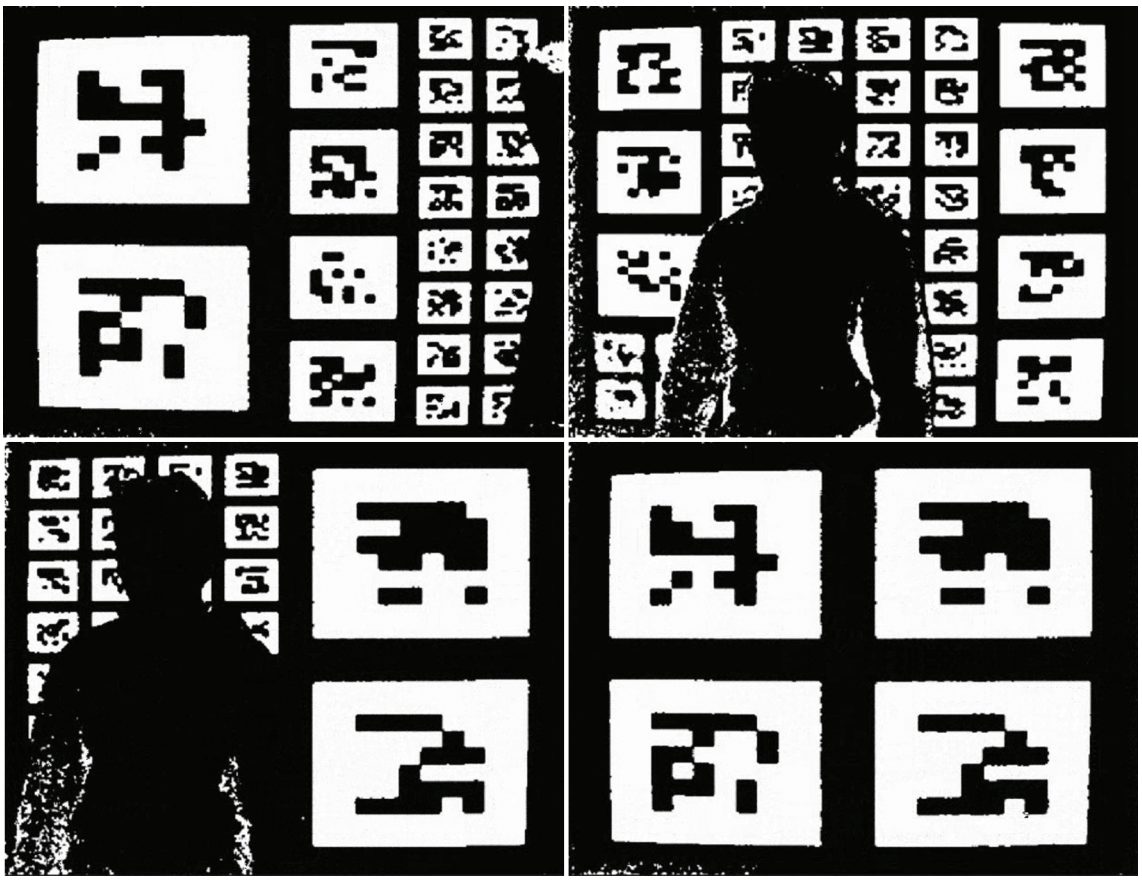


Figure 4.12: Example of the automatic marker placement: depending on the position of the moving person in front of the projection screen, the reconstructed markers adapt their size to ensure optimum visibility from the perspective of the observing camera.

4.2.3 Real-Time Implementation

Our software prototype for real-time processing of live recordings is implemented in C++ and OpenGL [149]. The complete image analysis [19] and the processing to generate the embedded code, its extraction and reconstruction, alpha matting and the marker placement is implemented entirely as a series of multi-pass renderings on the GPU using programmable vertex and fragment shaders. Details on the GPU based calculation of the spatial contrast and luminance information can be found in [67]. For the generation of the embedded code patterns, as well as their application for optical tracking, the ARTag library [51] was used.

To guarantee a constant switching between both Δ-coded images at a fixed frame rate of 120 Hz, quad buffer rendering in combination with a stereo-enabled DLP projector is required which en-

[19] except the optional optical flow calculation that was realized by using OpenCV[120]

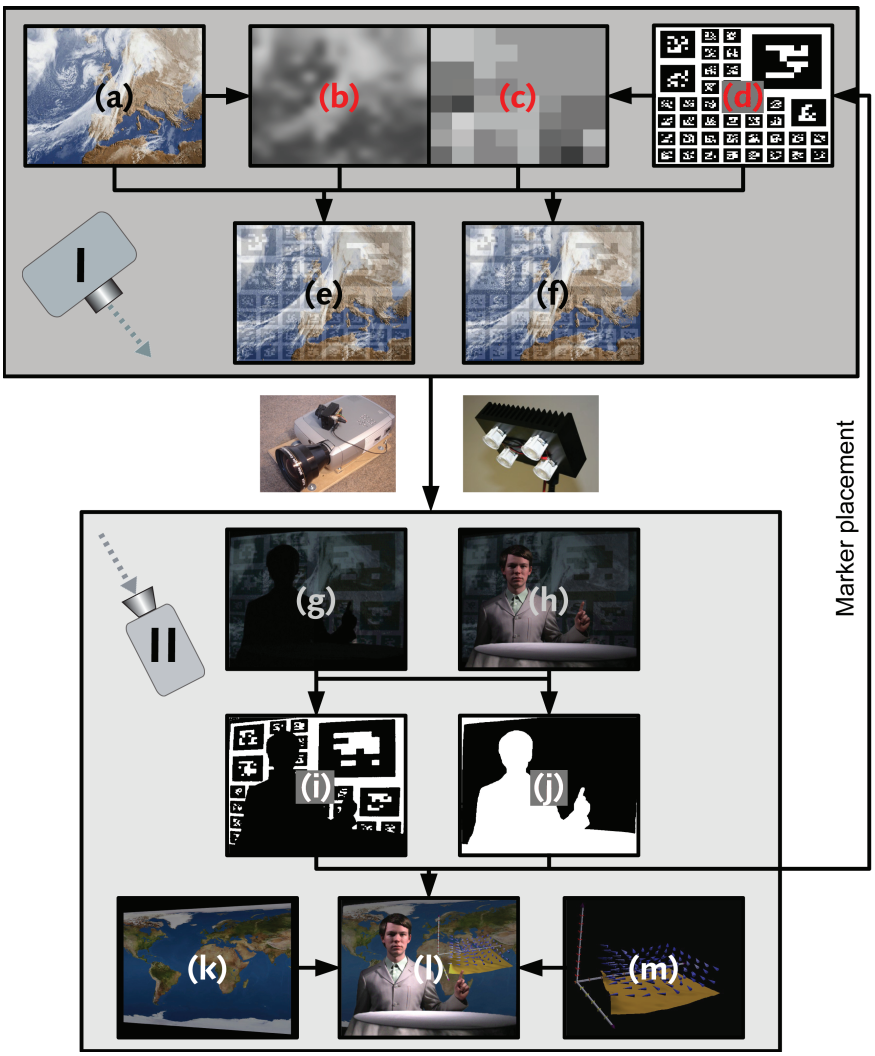


Figure 4.13: Flow diagram of the processing steps for the real-time system. Note that the camera-related and projector-related processing steps can be implemented in parallel. only the current marker placement has to be updated to adapt the markers to the current camera pose.

ables a constant projection of the complementary images at a refresh-rate of 120 Hz independently of the actual processing time of the system. The response functions of the projector and the camera were measured in advance. This information is necessary to generate linearized intensities of projection and camera images to realize a correct integration and reconstruction of the imperceptible code. By applying a gamma correction to the displayed image before embedding the code, the linear response of the projection device does not reduce the visual quality of the content presented. The DepthQ projector used in our prototype system does not provide sufficient brightness [20] for

[20] 1100 ANSI–lumen

professional studio usage, but it proved the potential of our proposed method. A professional usage, for example in television studio environments, could be realized by using projection devices with a higher luminous intensity.
Instead of activating the demosaicing functionality offered by the camera driver, we implemented a pixel grouping demosaicing algorithm [99] that is optimized for reducing color seams at intensity boundaries. This method offers a good trade-off between quality and performance. The algorithm is implemented as a multi-pass rendering of a series of vertex and fragment shaders on the GPU and delivers an enhanced quality at significantly higher frame rates compared to the camera driver's internal CPU based demosaicing algorithms. The LED system for flash keying was mounted in front of the projection screen and physical apertures were used to avoid direct illumination of the screen which would lead to the destruction of the embedded code (cf. figure 4.10). Care has to be taken to entirely illuminate the foreground objects to prevent the appearance of completely shadowed regions. To avoid misclassifications due to remaining shadows, a low keying threshold is used to classify very dark pixels, below the black level of the projector, as belonging to the foreground objects.

processing step	**ms**
Calculation of blending steps	3
Δ calculation	35
Marker embedding and display	0.5

Table 4.1: Projector-related performance of the prototype

Tables 4.1 and 4.2 summarize the durations of all relevant processing steps [21]. Note that tracking and the delta calculation are processed in parallel running threads which increases the overall frame rate. The overall frame rate of the prototype system reached 18 Hz. Using more powerful hardware components, however, would lead to a significant speed-up of the complete processing system.

We used the ARTag library [51] for optical marker tracking. Because of the fact that this application requires rectangular markers, care has to be taken to generate an undistorted projection onto the back-projection screen. While this can be achieved by placing the projector at a suitable position or adjusting the projection with the projector's internal adjustments, homography-based registration, as described in chapter 2.1.3.1, can also be used when other solutions fail, for example, due to space constraints.

[21] cf. section 4.2 for an overview of the used hardware

Our current prototype supports only one projector. An expansion to multiple projectors by using a distributed, synchronized rendering system can easily be realized due to the fact that only the calculated visibility tree has to be shared between the camera and the different projector modules.

Figure 4.13 summarizes all computation steps: The projector dependent steps are illustrated in the upper part (I) while the camera dependent steps are summarized in the lower part (II). The input image (a) as well as the current code image (d) is analyzed for its spatial frequencies and local average luminance values. From the results the Δ-values (b) and the local blending steps (c) are computed. Both are used to compute the Δ coded image pairs (e+f). These images are projected sequentially at a speed of 120 Hz while the foreground is flashed by the synchronized LED illumination system. The imperceptible code (i) as well as the alpha matte (j) are calculated from the image pair captured by the synchronized camera (g+h). The result is used to estimate the pose using ARTag and to compute the new optimized marker placement for the next frame as described in section 4.2.2. Knowing the actual camera pose as well as the foreground makes it possible to integrate a virtual background (k) and 3D objects (m) into the final image (l).

processing step	**ms**
Demosaicing	9
ARTag tracking	45
Adaptive marker placement	4
Matting/composite generation	8

Table 4.2: Camera-related performance of the prototype

4.3 Summary and Discussion

In this chapter we presented an imperceptible embedded Δ-coded projection that, in contrast to previous work, takes into account parameters of human perception for the optimal encoding and decoding of integrated patterns. It applies high-frequency temporal image modulation, and does not lead to a reduction or a non-uniform fragmentation of tonal values and intensities. It can be applied with standard projectors and does not require advanced calibrations other than a linearization and a gamma correction.

An analytical function for computing the optimal code contrasts used for coding, decoding, and code transitions was derived from a user study, and validated in a subsequent user evaluation. This

function is used to analyze the image content and the code patterns to adapt local Δ-values on the fly. Despite the integration of static code patterns, a way to efficiently exchange the code was described. The maximum number of locally required blending steps are computed automatically from image parameters of the projected content.

We demonstrated a real-time flash keying approach in combination with our coded projection for foreground extraction. By combining both techniques in a proof-of-concept prototype of a TV studio mock-up, a dynamic multi-resolution marker method was introduced that ensures a continuous in-shot camera tracking, despite possible occlusions and individual camera perspectives.

While the coded projection and illumination, as well as the capturing process are synchronized at a speed of 120 Hz, projected dynamic content was presented at a frame rate of 60 Hz. The final image composition that includes tracking, keying, matting, and rendering of augmented content [22] was carried out at 10-20 frames per second on our current hardware (depending on the number of detected markers and adjusted video resolution, the ARTag library required the corpus of 20 ms-60 ms for processing). This is clearly not yet sufficient for professional applications. Preliminary experiments showed that distributing the different steps to multiple PCs leads to a significant speed-up. If the tracking data is shared among a PC cluster, high-performance frame-grabbing enables the efficient exchange of high resolution image data.

As explained before, the flash keying technique we implemented can be combined with depth-of-field based keying, such as in [136], to support stable real-time matting. Furthermore, the tracking performance and quality needs to be improved significantly for professional applications (e.g. for TV productions). Since our approach is widely independent of the utilized marker tracking library, further investigations have to be carried out to experiment with alternative solutions such as ARToolkit [84], ARToolkitPlus [169] or Bazar [96]. At the moment, our system is limited to the performance and the precision of the ARTag library (see [51] for details). Currently, we support on-line and off-line augmentations. In the latter case, the captured images *I* and *I'* are only recorded to disk during run-time. During a post-production step, tracking, keying, matting and rendering can be carried out at a much higher quality level.

Although, our informal user study confirms the general validity of our approach and the experimentally derived coding principles, more complex user studies have to be carried out with respect to visual attention effects for animated content. This would possibly facilitate to increase the Δ-values

[22] i.e., foreground / background / 3D / composite

in animations. All of the techniques described were implemented directly on the GPU to achieve interactive frame-rates.

A technical challenge in a TV studio context will also be to adapt current standard studio camera technology to support fast-capturing and synchronization. Today, such cameras are synchronized to external displays via the standard BlackBurst signal generators, such as the Blackmagic Design Mini Sync Generator [61] at a speed of 50 Hz for PAL or 60 Hz for NTSC. Thus, the capturing at a field rate 60 Hz would decrease the extraction of the embedded code patterns to a maximum speed of 30 Hz. The projection speed and consequently the perception characteristics, however, are not effected by slower cameras (e.g., if only every third of the 120 Hz projected frames is captured with 40 Hz). Future projector generations will certainly provide even higher frame rates, as first 190 Hz projectors are already available on the market [158].

In the long term, we envision the combination of projector-based and LED-based illumination in modern television studios [15] and production studios. As it was shown, together with appropriate image correction techniques, such as geometric warping, radiometric compensation, and photometric calibration, this holds the potential to display imperceptible code patterns, such as the markers used for camera tracking, which are integrated into pictorial content or into the projected illumination spatially anywhere within a television studio. Temporally coded projector-based illumination would also support ad-hoc synthetic re-illumination as already shown in the small scale [14, 135]), and the extraction of depth-information (similar to that explained in [164] and [171].

The proposed system is capable of imperceptible, spatial and temporal code projection which allows the live-composition of perspectively correct augmentations over the recorded video stream. While the prototype showed the potential of synchronized projector-camera-lighting systems for real-time systems, these kinds of systems can be also used to provide a series of other, even more sophisticated techniques for high-quality off-line post-production. These are described in detail in the following chapters.

5 Projection Based Digital Video Processing Within Arbitrary Environments

The real-time compositing prototype for imperceptible pattern projection and flash keying presented in the previous chapter shows the potential of synchronized, modulated high-speed projection in the context of view-dependent live-video augmentations. The resulting visual quality of the implemented prototype for real-time compositing cannot, as yet, compete with professional post-production effects currently in use today. We therefore decided to investigate the possibilities of our basic idea of spatial and temporal intensity modulation for high-quality, off-line post-processing applications, which, except for augmented composites required for live-broadcasts, is sufficient for film productions and offers greater flexibility. In this chapter, we present a system that uses projectors as well as LED lighting to alternately capture a variety of information encoded within the illumination which can be used for different compositing effects as well as in-scene hints for moderators.

The basic idea of the proposed method is the synchronization of high-quality film cameras and analog LED lighting with off-the-shelf video projectors. This is combined with radiometric compensation, which makes it possible to project flexible keying patterns and other spatial codes on arbitrary real-world surfaces. The fast temporal multiplexing of coded projection and flash illumination facilitates professional keying, environment matting, the display of moderator information, scene reconstruction, and camera tracking for non-studio film sets without being limited to the constraints of a virtual studio. Blue screens and chroma keying technology are essential tools for digital video compositing. Professional studios apply tracking technology to record the camera path for perspective augmentations of the original video footage. Although this technology is well established, it does not offer a great deal of flexibility. For shootings at non-studio sets, physical blue screens can be installed and takes might have to be recorded twice, once with and once without blue screens, or parts have to be re-recorded in a studio, which significantly increases the overall production effort and constrains dynamic camera movements to reproducible, i.e. hardware-controlled, motions.

In this chapter we present a simple and flexible way of projecting corrected keying patterns and other spatial codes onto arbitrary diffuse surfaces using synchronized projectors and radiometric compensation. This widely neutralizes the reflectance properties of the underlying real surface. The temporal multiplexing between projection and flash illumination makes it possible to capture

the fully lit scene while still being able to key the foreground objects. Since the entire scene is recorded while no physical blue screen blocks the view, the footage of the full background scene can be used for video compositing. This means that recordings do not have to be made twice and keying is invariant to foreground colors. In addition, we embed other spatial codes into the projected images to enable camera tracking, environment matting, and the display of in-place moderator information. Furthermore, the reconstruction of the scene geometry is implicitly supported and facilitates special composition effects, such as shadow casts, occlusions and reflections. We propose a concept that combines all of these techniques into one single compact system that is fully compatible with common digital video compositing pipelines, and offers immediate plug-and-play applicability – a *"Virtual Studio 2 Go"* [66, 68].

5.1 Technical Approach

In this section, we describe the basic components of our system. We explain how temporal multiplexing of coded projection and flash illumination is technically implemented (section 5.1.1), and how projector-camera calibration and radiometric compensation is realized (section 5.1.2). Based on these components, the digital video compositing techniques that are outlined in section 5.2 are implemented.

5.1.1 Hardware Synchronization and Recording

We synchronize a high definition 3CCD camera with off-the-shelf DLP projectors and an Osram Ostar (5600K) LED lighting system [50]. This is being achieved by relaying the camera's composite signal directly to the G-sync channel of a quad-buffer graphics card [23] which can be triggered through an external source. The graphics card forwards the signal to the projectors and to the custom-built control electronics of the lighting system. Since the temporal synchronization has to be precise for every projected and captured frame, most LCD projectors as well as standard light bulbs cannot be used due to their too high latency [24]. Our current prototype system runs on a single workstation and records 720p frames at the HD scanning speed of 59.94 Hz. The uncompressed frames are captured via HD-SDI and recorded in *Yuv* 4:2:2[25] format to disk. Thereby, every even

[23] A Nvidia Quadro FX 4500 X2 G-Sync [119] in this case

[24] Please refer to chapter 7 for an in-depth description of all synchronization steps and the hardware details.

[25] In our case, this compression format ensures the required capturing rate when recording directly to disc. The luminance is sampled higher than chrominance in favor of quality alpha mattes that are derived from it.

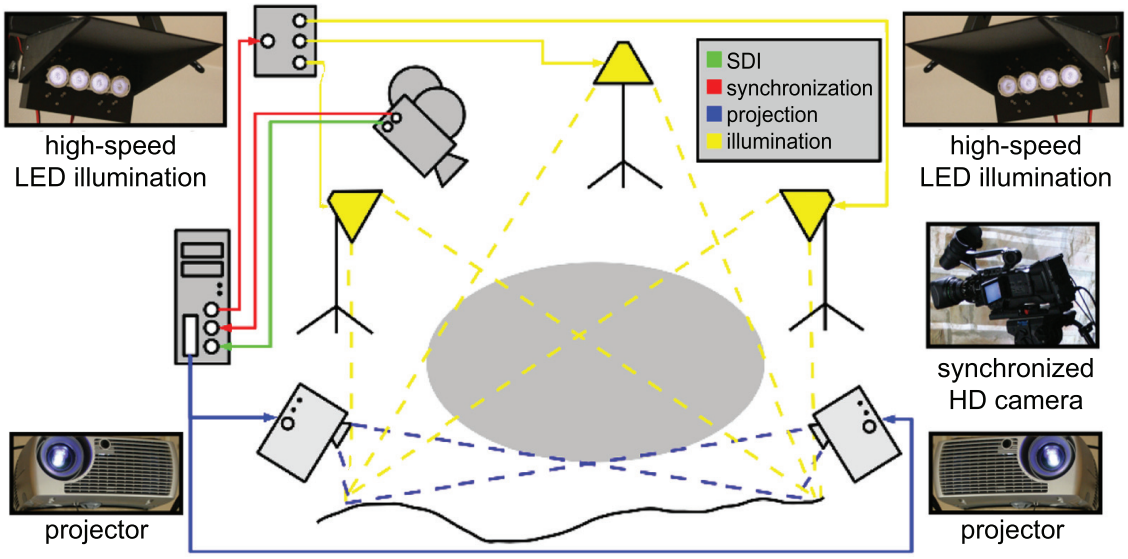


Figure 5.1: Schematic system overview of the synchronized components. The gray ellipse defines the space available for objects and actors that should be classified as foreground. The scene can be arbitrarily illuminated by the LED lighting system.

frame p is recorded with analog illumination turned off, while the background is illuminated by the projectors. Radiometric compensation, as described in section 5.1.2, neutralizes the possibly spatially varying reflection properties of the underlying material and enables the recording of distinct code patterns instead of the image of the background. Every odd frame i records the fully illuminated scene with the projection being turned off. Figure 5.1 gives an overview of the system setup as well as the required components.

The time-sequentially recorded frames are not registered during movements. Hence, after recording, the input frame sequence is first divided into two separate streams, one containing the i-frames while the other contains the p-frames. Both sequences are then doubled in their frame rate by generating the intermediate frames from the existing key frames via motion interpolation. We apply Adobe After Effects'^{TM}Time Warp^{TM}function for motion interpolation. After this step, p-frames together with the corresponding i-frames can be used for digital video compositing. In general, we encode keying information in the Y channel of p-frames, while the two chrominance channels, u and v, are used to encode additional code patterns to support, for instance, camera tracking and environment matting. This general principle is illustrated in figure 5.2. Details on our encoding techniques as well as on the required post-processing steps will be provided in section 5.2. Next, a short introduction to radiometric compensation and projector-camera calibration will be given.

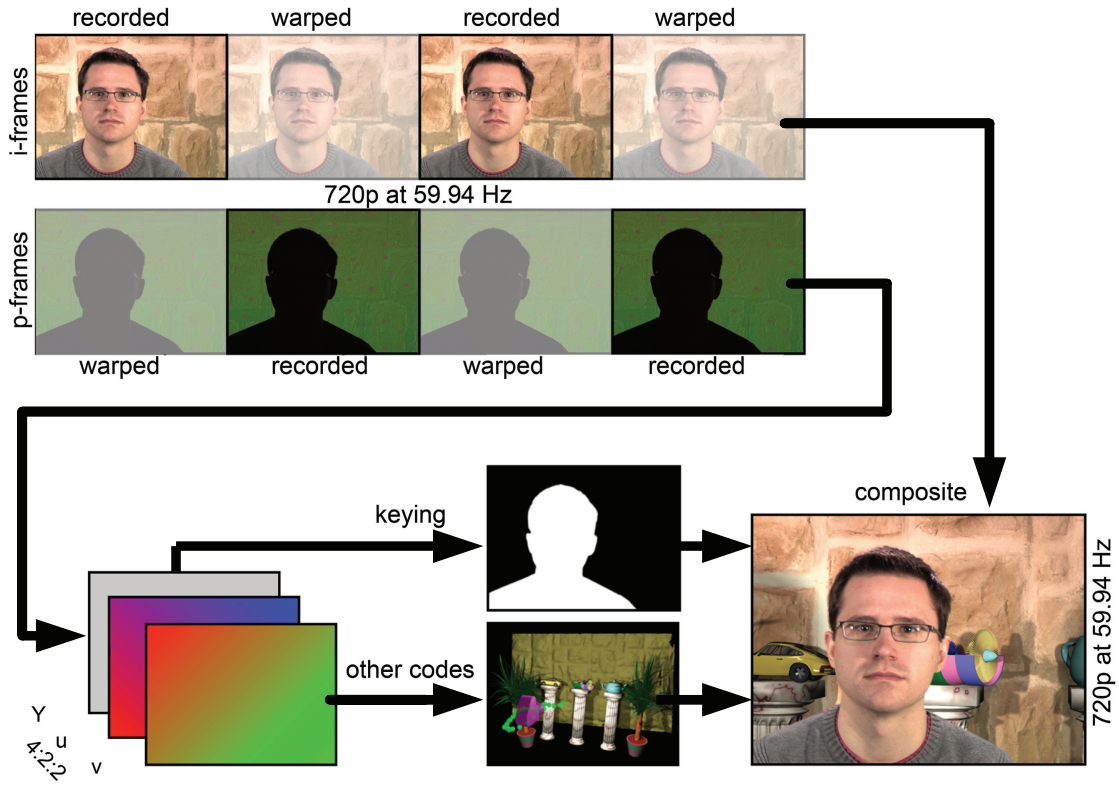


Figure 5.2: General encoding scheme of the proposed approach: Alternating i-frames and p-frames are recorded at HD scanning speed. After separating both frame types and reconstructing the missing frames via image warping, embedded codes can be extracted from the Yuv channels of the p-frames. Together with the i-frames and the reconstructed scene, they enable various digital video compositing effects.

5.1.2 Calibration and Radiometric Compensation

In our case, the projectors can be freely aligned as long as they do not project keying patterns onto foreground objects. Short-throw optics can be applied optionally – if they are required due to space limitations.

If it is not necessary to reconstruct the geometry of the background scene for supporting special compositing effects, like shadow casts (see section 5.2.2), then only the correspondence between projectors' pixels and surface pigments, as well as radiometric properties of the surface have to be determined. For each individual projector, this is achieved in the following way:

We are projecting a series of binary images encoded to generate a Gray code [18] and line stripes [62] onto the surface and capture it with the camera for measuring the projector-to-camera pixel correspondences as explained in detail in section 2.1.3.1. Thereby, the camera is aligned to cap-

ture a maximum of the projector's display region. After this step, the local radiometric properties are measured from the same camera position, as described by Yoshida et al. [186] (cf. section 2.1.3.2). After mapping these measurements from the perspective of the camera to the perspective of the projector, we can compute a 3x4 color transformation matrix V for each projector pixel. This matrix stores the surface reflectance and the environment light contribution at the surface pigment that projects onto the corresponding projector pixel as well as the amount of color mixing between projector and camera. The color mixing is due to the different spectral band-pass responses of the color filters used by the projectors and the ones applied by the camera. Note, that the projectors have to be linearized beforehand.
In case of diffuse surfaces and assuming local reflections only, the forward light transport between the projector I, over the surface towards the camera R can be approximated with $R = VI$. For compensating the modulation effects of the light on the surface, we compute the inverse $I = V^{-1}R$. This means, that if we want to capture the desired image I by the camera, we have to project the compensation image R onto the surface. In general, this per-pixel operation is referred to *radiometric or photometric compensation* [114, 186, 65]. If multiple projectors are applied, this calibration step is repeated for each unit separately, while overlapping projector regions have to be blended appropriately. For more details on radiometric compensation technique and blending, the interested reader is referred to the state-of-the-art summary presented in [16].
In case that the surface is not perfectly diffuse, the inverse light transport matrix T^{-1} can be used for radiometric compensation as explained in chapter 2.1.3.2. This allows the compensation of all local and global illumination effects for the initial perspective of the camera. For other perspectives, however, the light transport matrix is not valid and thus radiometric compensation will fail. If camera motion is required, we restrict our approach to projections onto mainly diffuse background surfaces, since the compensation of view-dependent modulation effects would require the known camera pose at runtime.

If, in addition to the radiometric surface parameters the scene geometry is also required, we have the option to calibrate the intrinsic and the extrinsic parameters of all projectors and of the camera at different positions with the help of a temporally aligned printed ARTag [52] calibration pattern. Having these parameters as well as the projector-to-camera pixel correspondence determined from a Gray code projection, as explained above, we can reconstruct the geometry of the static background scene. This is currently limited to mainly diffuse surfaces. An example of a composite using the geometry of a reconstructed natural stone wall to simulate consistent shadow casts is illustrated in figure 5.4.

5.2 Digital Video Compositing Techniques

The subsequent steps are based on the techniques explained in section 5.1, and are fully compatible with common digital video compositing pipelines. In fact, we apply professional visual effects software tools for most steps, such as for keying, 3D rendering, motion interpolation and match-moving after the recored frame sequences have been pre-processed. This offers an immediate plug-and-play applicability of our proposed concept.

5.2.1 Keying

As mentioned above, we encode keying information into the *Y* channel of our *p*-frames – approaching a uniform luminance on the background surface. Since the *p*-frames are radiometrically compensated before being projected out, locally different surface reflectance properties are widely neutralized. Recoding the *p*-frames and separating the *Y* channel will then show a consistent intensity at the background while the foreground objects are unlit, and therefore appear black. Conventional luma keyers (cf. section 2.2.1) can now be applied for generating high-quality alpha mattes. This is illustrated in figure 5.3.

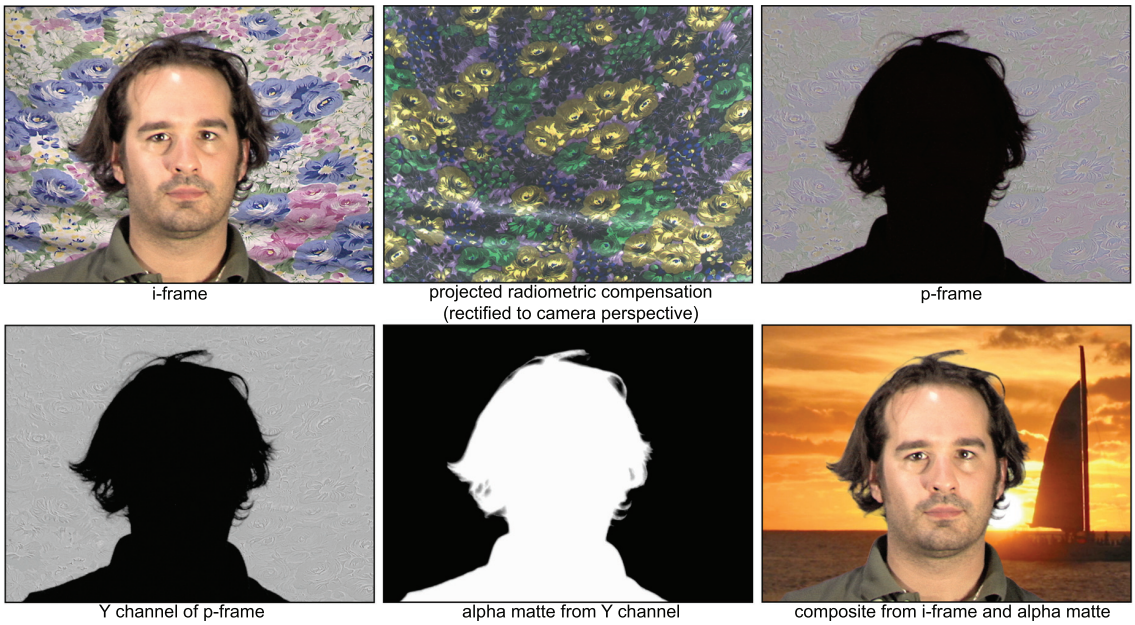


Figure 5.3: The radiometric compensation makes the background surface appear with a widely uniform luminance in the Y channel of the p-frames. This allows producing high-quality alpha mattes – even if the background surface's texture is fairly complex as shown in this example.

Figure 5.4: Using the projector-camera system to reconstruct the projection surface (a) makes it possible to correctly simulate shadow casts between real and virtual parts of the composite (b).

Assigning a chrominance to the luminance values in the Y channel makes the input images compatible to chroma keyers as well. The keying information is not changed with this, but todays chroma keyers offer more functionality than luma keyers. Note, that in contrast to conventional chroma keying in combination with blue screens, our method is invariant to foreground colors. One other advantage is that the entire scene is recorded in the i-frames while no physical blue screen blocks the view onto the background scene. Thus the footage of the full background scene can be used for video compositing. We apply dvMatte pro^{TM}[49] for keying.

5.2.2 Scene Reconstruction

As explained in section 5.1.2, we can measure the extrinsic and intrinsic parameters of projectors in the course of our initial calibration. Having these parameters allows reconstructing the scene depth of the static background surfaces. Together with the measured radiometric surfaces parameters, we can use this information in the final compositing process for creating special effects, such as shadow casts from synthetic objects onto real surfaces, reflections of real objects in digital ones, or occlusions of virtual objects in real ones. We use Autodesk Maya^{TM}2008 Unlimited for rendering the final composite. Furthermore, the geometry of the background scene is required for handling transparent foreground objects correctly for applications that require moving cameras, as described in section 5.2.3. Figures 5.4 and 5.5 show examples of synthetic shadows cast by virtual objects onto the real surface and vice versa.

Figure 5.5: Another example composite. Note the shadows which are cast correctly from the augmented turbine onto the real walls.

5.2.3 Environment Matting

The computed alpha mattes as described in section 5.2.1 alone, do not support the correct keying and compositing of transparent objects. Therefore, the refraction of light through such objects cannot be simulated in combination with a synthetic background. In general, this is enabled through a technique referred to as *environment matting* [191], which is in principle an extension to alpha matting.

Different approaches exist for generating high-quality environment mattes. Most of them, however, require one or multiple displays in the background for presenting Gray codes or intensity codes [127, 189, 33] [26]. Therefore, multiple images must usually be captured for calibration and dynamic objects are not supported. These limitations make such techniques useless in our context, since we need to produce environment mattes for static and for dynamic objects from a single frame in an arbitrary environment.
We apply the technique described in [31]. This method allows the acquisition of environment mattes from a single frame. It is restricted to non-colored, non-glossy transparent objects with possible specular highlights that do not cause significant chromatic aberrations. The coordinates of the static

[26] cf. section 2.2.1 of the related work chapter for a detailed explanation.

background environment are encoded into the u and v channels of our p-frames [27]. This does not influence the quality of the alpha matte, because this is pulled exclusively from the separated Y channel.

For static camera situations, u, v can directly map to the x, y coordinates of an initially recorded background image. The code image, which possibly contains additional keying information in the Y channel, is geometrically warped to the perspective of the projectors, radiometrically compensated and displayed, as explained in section 5.1.2. Recoding this projected code through a transparent object allows estimating the optical transformation caused by refraction of each camera pixel to the background image.

Supporting environment matting for moving cameras requires knowledge about the camera's pose and about the geometry of the background scene. Both are information that we are able to retrieve, as explained in sections 5.2.2 and 5.2.4. For moving cameras, the initially recorded background image as well as the computed u, v reference code are texture-mapped onto the reconstructed scene geometry. From new perspectives of the known camera's pose, the scene geometry is rendered together with these two textures. With both perspectively corrected images we can proceed as explained above for a static camera. As described in [31] the resulting environment matte is smoothened with a non-linear filter to remove noise in the calculated matte, but instead of using anisotropic diffusion, a bilateral filter [161] was used to speed up the processing time. This filter is applied spatially, but also temporally to guarantee temporal consistency.

As illustrated in figure 5.6, environment matting should only be applied to the image area that is covered by transparent objects, since the resolution of the captured ramps is less than the actual image resolution. Camera noise and limitations of radiometric compensation reduces the quality of the ramps even further. We key out opaque foreground objects first in each recorded i-frame as described in section 5.2.1. The image area of these objects is not considered for the following steps.

For keying the transparent objects that are located in front of the background, we compute the $L2$-norm difference image between the pre-masked content in i-frames and a reference image that shows only the u, v ramps being projected onto the background surface (without foreground objects). As explained above, this reference image can either be directly captured for a static camera, or is rendered for the correct perspective of a moving camera. Image transformations that result from refraction at transparent objects should lead to large differences. The pixel with the smallest

[27] A maximum number of chrominances that are not clipped in the native RGB color space of the projector can be achieved with Y=0.5

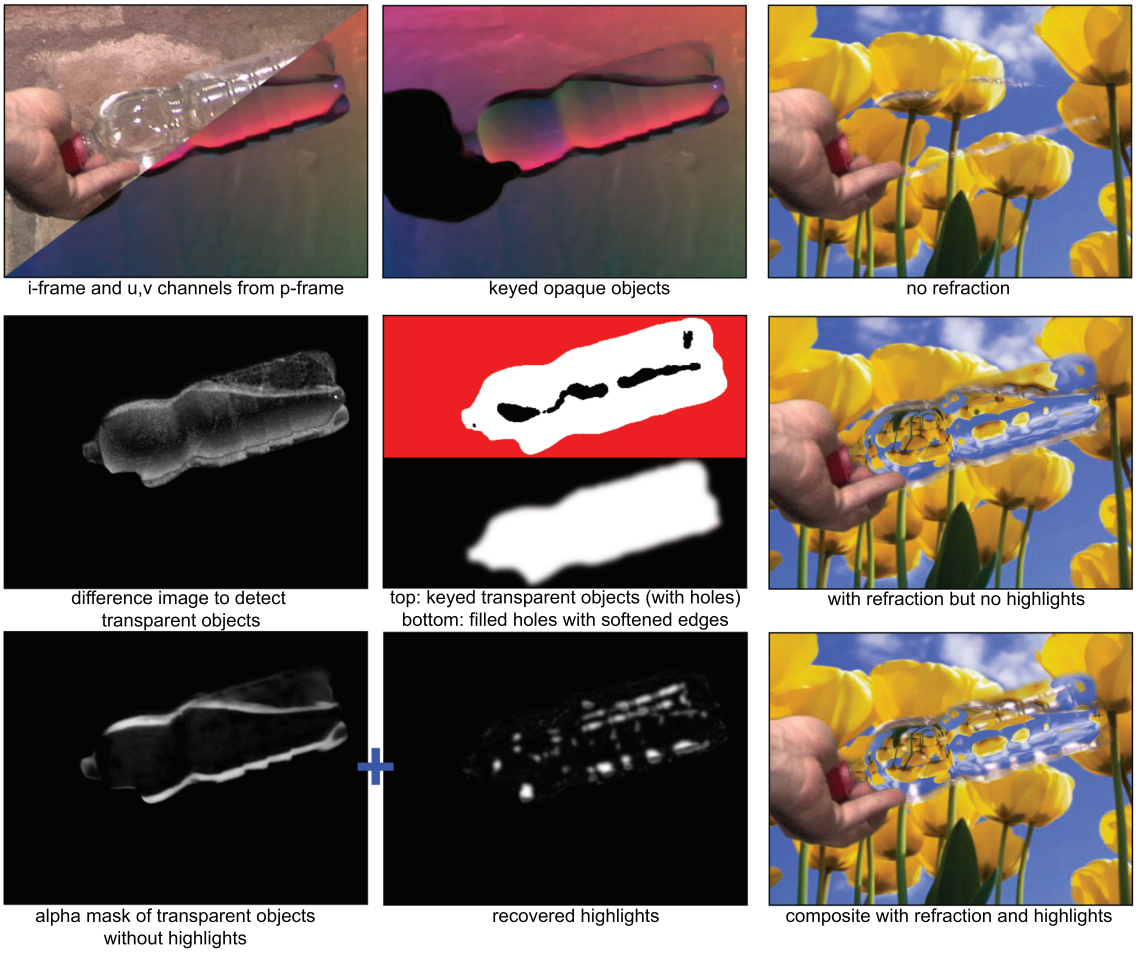


Figure 5.6: Encoding intensity ramps in the u,v channels of the p-frames for environment matting allows keying transparent objects and simulating refraction effects. Highlights that do not appear in the initial alpha matte can be reconstructed and added.

difference is most likely not being refracted, and its position is stored. The difference result is first median filtered to eliminate outliers. Then it is thresholded to remove clearly non-refracted pixels. During this operation, however, pixels of transparent objects that are not refracted much could be removed too. We will potentially lose specular highlight information with these pixels. To avoid this, we do not apply the thresholded difference image directly for masking. Instead, we flood fill it with a keying ID from the stored position of the pixel with the smallest difference to approximate the silhouettes of the transparent objects. All pixels that are not filled with this ID resemble the final keying mask. It is finally Gauss filtered to smoothen remaining noise at edges. The pixels of the u, v ramp inside the mask are used for environment matting, as explained above. This process is visualized in illustrated in figure 5.6.

Discontinuities on transparent objects, such as holes through which the background is visible surrounded by refracting material or objects with little refraction at their silhouette edges would lead to misclassifications. These cases are rare, but they clearly represent a limitation of our approach. Specular highlights on transparent objects that are located in front of background surfaces are still lost. The reason for this is, that they are not produced in the *p*-frames initially (i.e., when lighting is turned off), and the region where they should appear is keyed as background. We recover them, similar to the approach presented in [87], by searching pixels whose intensities are above a defined threshold in the masked image regions of *i*-frames that contain transparent objects. The luminance of the discovered highlight pixels are added to the alpha mask which is used for keying. They can be optionally Gauss filtered for smoothening their edges in case that an optimal highlight threshold is difficult to adjust due a possible high brightness of the remaining pixels.

5.2.4 Visual Hints for Camera and Moderator

Match-moving methods support tracking of the camera if an adequate quantity of static background features can be detected in the recorded *i*-frames. In particular for non-textured background surfaces, this is not always given and match-moving will fail in these cases. One apparent possibility of our approach is to project synthetic features if not enough natural features are present.

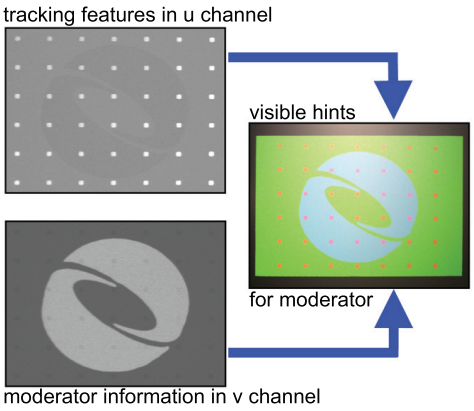


Figure 5.7: If transparent objects do not have to be supported, tracking features can be encoded directly into one chrominance channel while the other chrominance channel is available for additional information, such as visible moderator information.

If transparent foreground objects do not have to be supported as described in section 5.2.3, we can encode these features directly in either the *u* or the *v* channel of our *p*-frames. Then the other channel can then be used to encode visual hints for moderators, for example. The missing in-place moderation information that would support a more natural interaction in blue boxes is one of the challenging problems to be solved for virtual studios. Since in our case, this information appear mixed together with the data encoded into the other two channels, the image quality of presented visual hints is not high. But it might be sufficient for displaying simple and dynamic moderation information. An

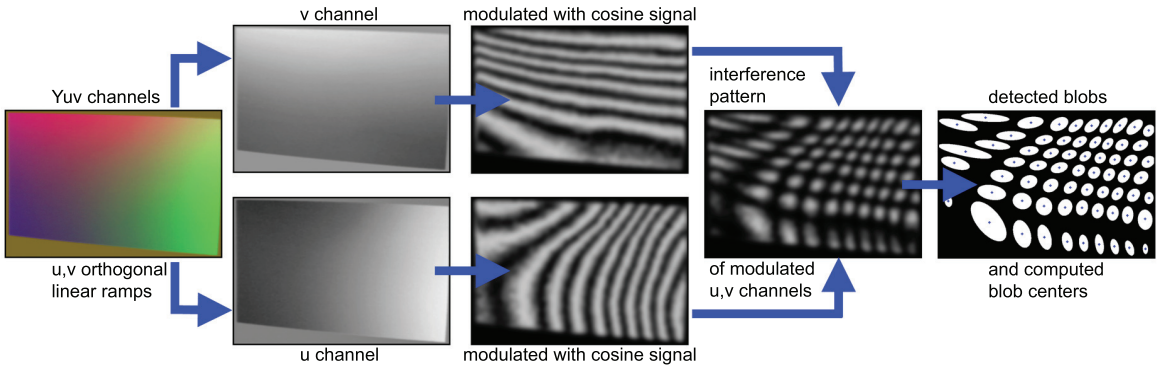


Figure 5.8: The same u,v ramps that are used for environment matting can be applied for camera tracking. The orthogonal ramps are modulated with a cosine signal. The interference pattern of these two signals remains constant on the background surface and allows computing static tracking features.

example is shown in figure 5.7. This idea is similar in spirit as the technique described by [58]. In our case, however, such a method is seamlessly integrated into an approach that supports camera tracking and keying simultaneously.

If transparent objects do need to be supported, the u,v channels are occupied with the reference ramps. In this case, no visible moderator hints can be embedded. As shown in figure 5.8, however, useful features for tracking can still be extracted. Theoretically, the u,v coordinates that are used for environment matting are unique and therefore could directly be used to index individual tracking features. In practice, these features would not be stable due to the relatively low resolution of the u,v ramps as well as due to blur and noise that results from limited camera and projector responses and depth of field. Instead of addressing tracking features with individual u,v coordinates, we modulate the linear u,v ramps with additional signals in the captured image. We chose cosine functions for a frequency modulation in both orthogonal directions. For normalized u,v coordinates, the interference pattern of both modulated signals is then computed with

$$A * cos(2\pi f_1 u + \theta_1) * cos(2\pi f_2 v + \theta_2) \tag{5.1}$$

where A is the amplitude of the resulting signal, f_1, f_2 are the frequencies in u and in v directions, and θ_1, θ_2 are the phase shifts in u and in v directions. These parameters can be used to adjust the interference pattern for aligning it optimally to the background surfaces. After capturing a p-frame, extracting the u,v ramps, and computing the interference pattern, the result is Gauss filtered to reduce noise. Then areas of constructive interference are found via blob detection, and each area's center point is determined by averaging the weighted sum of interference signals within its blob

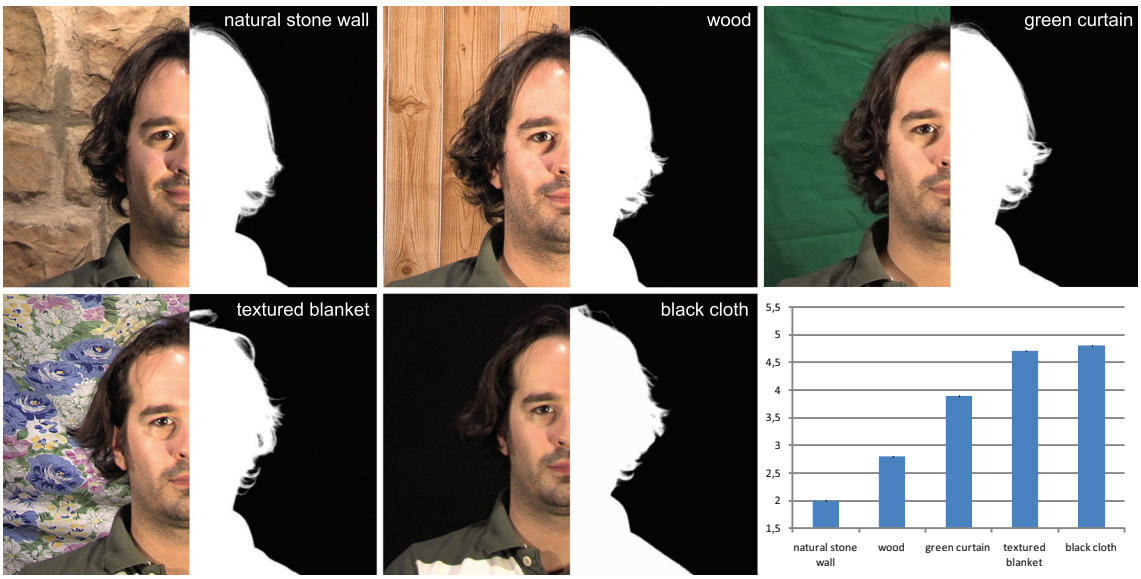


Figure 5.9: While the keying quality remains fairly constant for different background surfaces, the encoded u,v ramps used for environment matting become more distorted with complex textures and colors, or against dark surfaces. This is shown in the diagram in the lower right: it shows the average increase in displacement error of the environment matte after modulating the projected color ramps with the background surface with respect to the unmodulated case (x1.0).

region. The center points are used as tracking features. This is quite invariant to noise, defocus and low ramp resolutions.

In both cases, we use Autodesk Maya^{TM}2008 Unlimited's Live^{TM}module for match-moving based on synthetic feature points. While this semi-automatic software requires minimal user-input, other solutions [1] are able to generate the complete match-moving process fully automatically.

5.3 Summary and Discussion

In this chapter we presented a flexible hardware prototype of a *"Virtual Studio 2 Go"*, a synchronized system using temporal and spatial light modulation to generate various digital video compositing effects via post-production. In combination with radiometric compensation, the systems allows to project patterns for alpha matting, environment matting, camera path reconstruction and moderator hints directly onto arbitrary surfaces. This allows to use every pixel of the recorded footage to be used in the final composite – no physical blue screens or green screens are blocking parts of the scene.

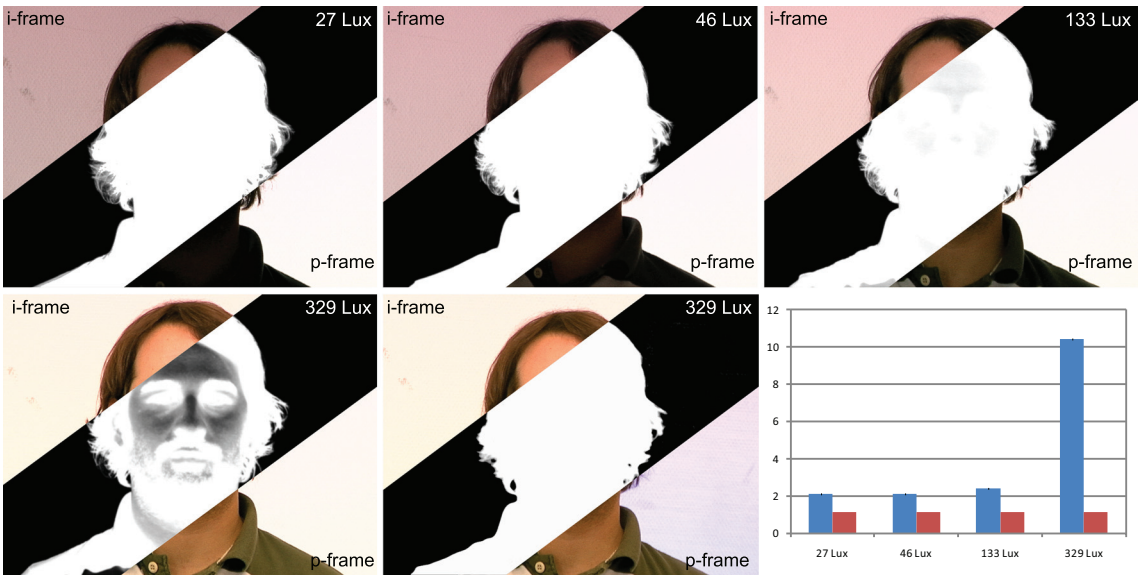


Figure 5.10: The keying quality decreases with increasing environment light. This is mainly due to the limited brightness of the projector. Using chroma keying instead of luma keying for bright environments improves this situation (image center bottom). The diagram on the bottom left plots the average noise radius of the tracking features on the camera's image plane in pixels: since the encoded u,v ramps for environment matting are washed out in bright ambient light conditions, tracking features computed from them become more noisy and the tracking quality decreases (blue bars). Synthetic features that are directly encoded into the p-frames, however, remain quite robust with increasing levels of ambient light (red bars).

One obvious limitation of the techniques described in sections 5.2 when applied in our context is, that they will become less robust with an increasing amount of uncontrollable environment light, or with radiometrically complex background scenes. If the environment light is too bright or the background surface absorbs too much light, radiometric compensation will fail and the quality of keying, environment matting and tracking will decrease. This is mainly due to the limited brightness and the limited dynamic range of the projector. Figure 5.9 illustrates the quality for various background surfaces and in figure 5.10 an impression of the quality under the influence of increasing environmental illumination is given.

While keying remains robust in dark environments with controlled flash illumination – regardless of the background surface – it will fail more and more if the brightness of the environment light approaches or exceeds the brightness of the projector. The u, v ramps that are used for environment matting and tracking are also distorted on complex colored surfaces, or are washed out in bright

environments. Thus, environment matting and tracking will become less precise with increasing environment light or complex backgrounds. As a rule of thumb, we found that if the brightness of the projector(s) is not less that 30% of the environment light, keying, environment matting and tracking can be performed in an aceptable quality. Since synthetic tracking features do not have to be generated on textured surface portions (they provide enough natural features for match-moving) which would potentially distort the u, v ramps, the tracking quality is also rather invariant to the complexity of the background surface – unless the background surface does not reflect a minimum amount of light anywhere. If tracking features are directly embedded into the p-frames rather than computing them from the u, v ramps, the tracking quality will also remain constantly high (also for all cases shown in figures 5.9 and 5.10, including the green/black backgrounds).

For bright environments, we therefore propose to project p-frames with a uniform keying color to enable true chroma keying rather than using the Y channel for luma keying. In this case, tracking features can be directly embedded into the p-frames using a second keying color. Applying chroma keying twice (once for each the two keying colors) will result in a robust alpha matte (by adding both keying results) and in stable tracking features. This is shown in figure 5.10 (lower-middle image). In this situation, however, other information cannot be encoded, environment matting is not supported, and keying is not completely invariant to foreground colors anymore.

We believe that all explained techniques can be carried out in real-time when being completely implemented on graphics hardware, or when being forwarded to existing video compositing hardware. The integration of image-based relighting techniques belongs also to our future investigations.

6 Color Invariant Chroma Keying and Color Spill Neutralization for Dynamic Scenes and Cameras

The system described in the previous chapter facilitates the generation of various composition effects outside of studio environments by projecting radiometrically compensated images onto arbitrary lambertian surfaces. While this approach provides high flexibility in terms of scene setups, it requires a completely synchronized lighting environment to generate the maximum quality in terms of alpha and environment matting. In contrast to this, we finally present another synchronized illumination-camera system for high-quality alpha matting that can be used to overcome the two main limitations of standard chroma keying: its color dependency implications for foreground colors as well as the undesired effect of color spill. The system can easily be integrated into a standard studio production environment and presents a straightforward solution to solve the mentioned problems.

Keying or matting is one of the most fundamental tools for image-, broadcast-, and film-production. It makes it possible to separate the foreground from the background in captured images or recorded video footage and to compose new images or video sequences from the extracted foreground and a synthetic background. Many visual effects rely on high-quality keying. A large variety of keying techniques exist, such as luma and chroma keying, difference and depth keying, polarization and defocus keying, and natural image matting techniques. All of them have distinct advantages and limitations as already described in chapter 2.2.1. Today, chroma keying [166] is the most common method for generating professional alpha mattes by using colored, mostly blue or green, backdrops, typically in the form of painted walls or colored curtains. Its implications for foreground color (foreground objects with a color similar to the background cannot be keyed) and problems of color spill (i.e., visible color spreading from the green or blue background onto the foreground in the final composite) are the two main disadvantages of chroma keying.

In this chapter we present an enhanced chroma keying method that is based on the fast temporal switching of backdrop colors [69]. Compared to standard chroma keying, intrusive color spill is transformed into neutral white background illumination. Since the chosen colors sum up to white, the chromatic (color) spill component is neutralized when integrating over both backdrop states. In contrast to digital color spill suppression algorithms, our method is automatic, supports real-time rates, and reproduces all colors correctly. It is not variant to the distinct optical modulation of foreground materials, such as polarization, retro-reflection or infrared reflection. The ability to separate

both states additionally makes it possible to compute high-quality alpha mattes, but, as opposed to conventional chroma keying, it is invariant to foreground colors, enables ad-hoc adjustments of scene lighting during recording, and facilitates uniform background illumination. Our approach supports dynamic foregrounds as well as freely moving cameras, is fully automatic, and can be combined with camera motion tracking techniques. This combination of features is not provided by any other related matting technique described above. In this chapter, we explain different ways of realizing temporal backdrops and describe how keying and color spill neutralization are carried out, how artifacts resulting from rapid motion can be reduced, and how our approach can be implemented to be compatible with common real-time post-production pipelines.

The remainder of this chapter is organized as follows: Section 6.1 gives a preliminary overview over the proposed method. Our approach of temporal chroma keying will be explained in detail in section 6.1.1. A color seam correction required to eliminate errors emerging from the temporal recording will be presented in section 6.1.2. In 6.1.3, different realizations of temporal backdrops are described. Next, a practical implementation of the complete method is given in section 6.2. Results, limitations as well as a comparison to other color spill suppression methods are shown in section 6.3.

6.1 Color Invariant Chroma Keying and Color Spill Neutralization

Video frames which are recorded at a speed of, for instance, 30 fps can be exposed for up to a 1/30th of a second. During that integration time, captured reflectance and color spill are blended. If chroma keying is applied, the latter is less problematic for extremely fast camera or foreground motion, since it is mixed with traces of motion blur. Under normal conditions, however, it is clearly visible. Instead of integrating each frame at normal speed, we propose to halve integration time and, consequently, double the frame rate. Therefore, an adjustable, temporal backdrop illumination sequentially displaying orthogonal colors in the RGB color space has to be synchronized to the camera. Integrating two subsequently recorded sub-frames (C_a and C_b) digitally by averaging them via:

$$C_{ab} = (C_a + C_b)/2 \tag{6.1}$$

to return to the original frame rate neutralizes the chromatic (color) spill component. The reason for this is, that adding two saturated and orthogonal backdrop colors will result in a homogeneous illumination spectrum when averaged. This mimics a neutral backdrop illumination which also neutralizes color spill in the foreground. Contrary to the classical triangulation approach presented in

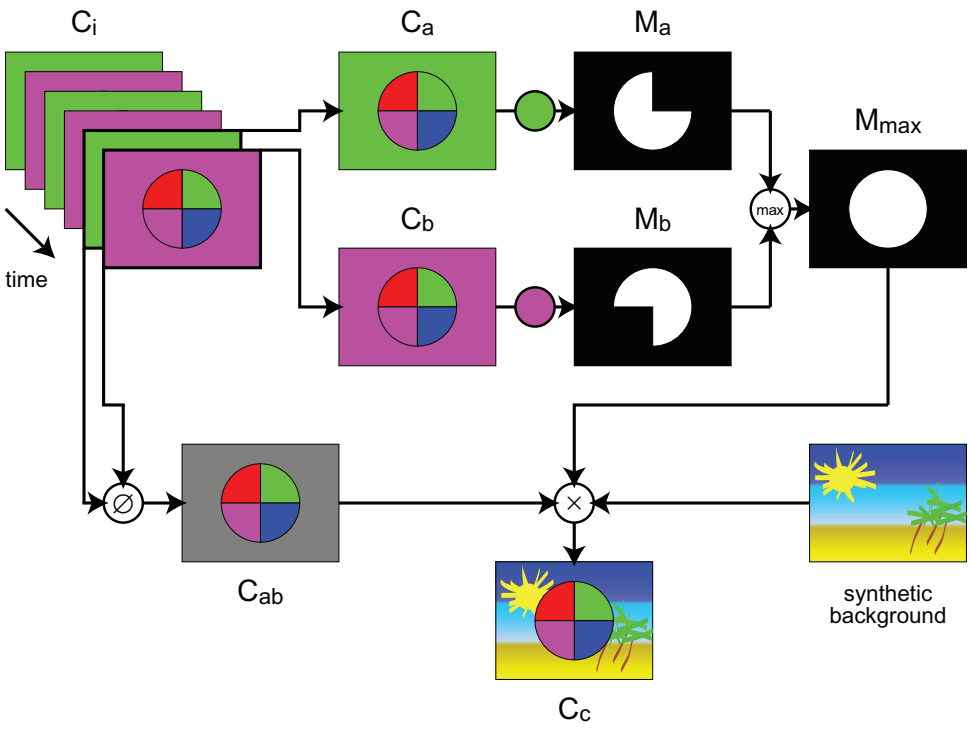


Figure 6.1: Schematic diagram of our approach: Sequentially recorded video frames C_i are separated into (C_a,C_b)-pairs according to the synchronously changing backdrop color. Chroma keying is applied independently to both frame sequences. The resulting alpha mattes M_a and M_b contain partially misclassified pixels when foreground and background colors match. The maximum ($M_{\max}$) of M_a and M_b is free of misclassifications. By blending the input frame pair into $C_{ab} = (C_a + C_b)/2$, color spill is transformed into a neutral background illumination. The final composite C_c is generated by applying $M_{\max}$ to C_{ab} and then combining the result with a synthetic background.

[151], each sub-frame is chroma keyed individually and the maximum of both alpha mattes is used. This process is illustrated in the schematic diagram in figure 6.1, and examples are shown in figure 6.2 and 6.4. Using the maximum of both mattes makes this approach insensitive to unregistered image pairs. The method of [151] which is discussed in detail in chapter 2.2.1 fails in this situation, as shown in figure 6.3, since it supports only static scenes and cameras. Instead of analyzing and suppressing spill in each frame like classical techniques [166, 48], our method neutralizes color spill primarily by integrating over two complementary spill contributions during recording. The result is equivalent to a neutral white background illumination. This requires temporal backdrops that can be switched at the speed of the camera's frame rate as explained in section 6.1.3.

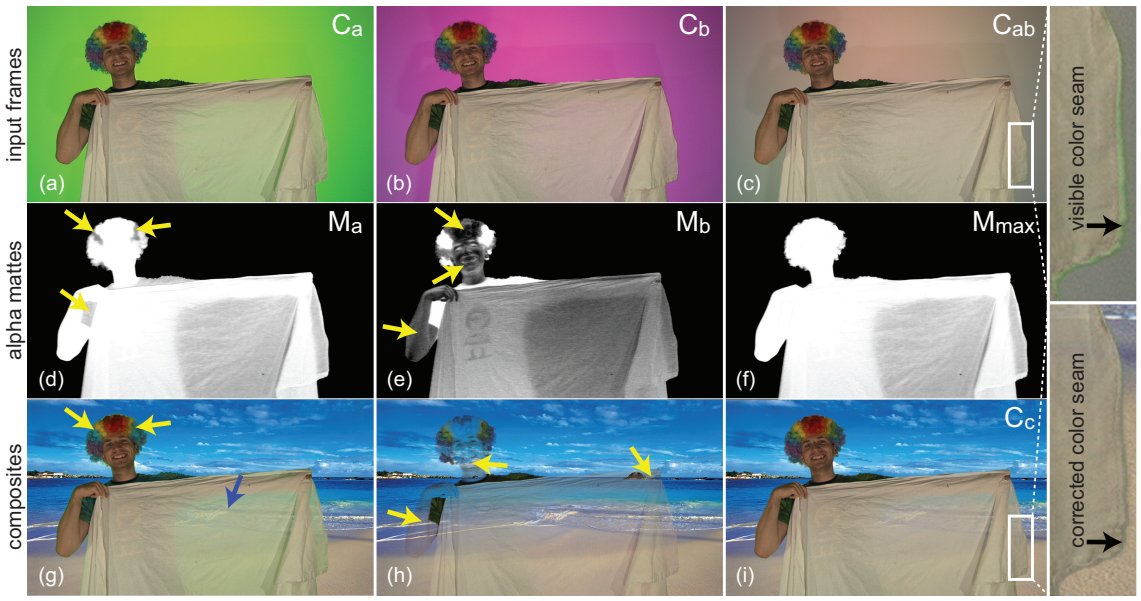


Figure 6.2: Chroma keying and blending the recorded image pair transforms color spill into a neutral background illumination (c) when complementary colors are used as backdrops (a), (b). The alpha mattes of (a) and (b) are generated individually (resulting in (d) and (e)) and are then combined to compute an optimized alpha matte (f). This neutralizes color spill and is independent of the foreground colors (compare (g) and (h) with (i)). Possible color artifacts during motion (close-up in (c)) are also corrected (close-up in (i)). Please zoom into individual image sections to see details of spill (blue arrows), keying (yellow arrows) and color seams (black arrows).

6.1.1 Temporal Chroma Keying

As explained above, we pull an individual alpha matte (M_a and M_b) from each sub frame (C_a and C_b). If the foreground color matches one of the background colors, then the corresponding matte will contain misclassified pixels. The classification of these pixels in the other matte, however, is always correct. Computing the maximum M_{max}=max(M_a,M_b) results in an alpha matte that does not contain misclassified pixels. This matte is then used to compute the final composite image C_c, as displayed in figure 6.1 and demonstrated in figure 6.2. Thus, our technique also becomes invariant to foreground colors, which is another relevant advantage over most classical chroma keying methods. This approach works well for static scene setups. However, if the content in C_a and C_b becomes misregistered during fast movements of the camera or the foreground, miscolored seam regions appear in C_{ab} and consequently also artifacts in C_c. This is displayed in figures 6.5a-c, and shown in figure 6.2c. An additional post-processing step identifies these regions and color corrects the artifacts by reconstructing the missing foreground information as shown in figure 6.2i.

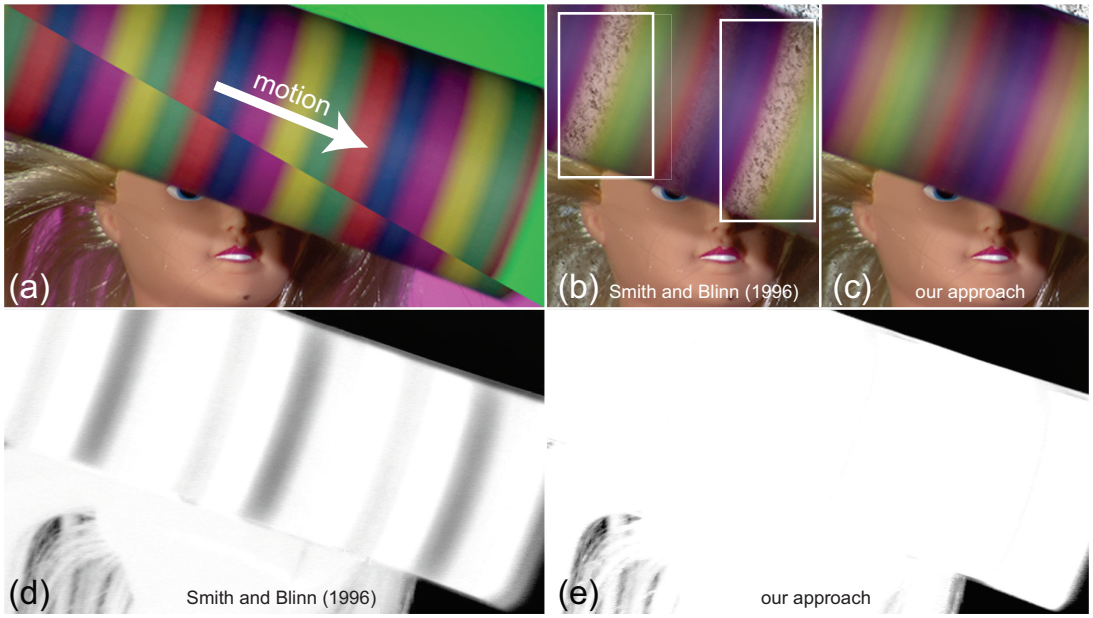


Figure 6.3: The method of Smith and Blinn [151] generates holes (d) and consequently imperfect composites (b) that result from misregistration during movement (a). These regions are classified correctly by our approach (c),(e).

6.1.2 Color Seam Correction

Figure 6.5 outlines our color seam correction approach: First, we binarize M_{max} to M_{maxb} (cf. figure 6.5e). Then we find the contours in M_{maxb} by applying a Canny filter. Next we extrude these by dilation, as shown in M_{cext} in figure 6.5f.

By capturing with 59.94 Hz, we assume that the maximum motion in two consecutive frames is less than the dilated region in M_{cext}. Since we are only interested in the contour neighborhood of M_{maxb}, the intersection between M_{cext} and M_{maxb} gives us the regions with potential color seams, as shown in figure 6.5g. For all pixels within this intersection, we need to determine if their seam colors are caused by the background in C_a or by the background in C_b, or if they belong completely to the foreground (pixel 4 in figure 6.6) or the background (pixel 1 in figure 6.6). For the latter two cases, no additional actions are necessary, whereas the first two cases require special treatment. These situations can be determined by comparing the absolute RGB color difference between C_a

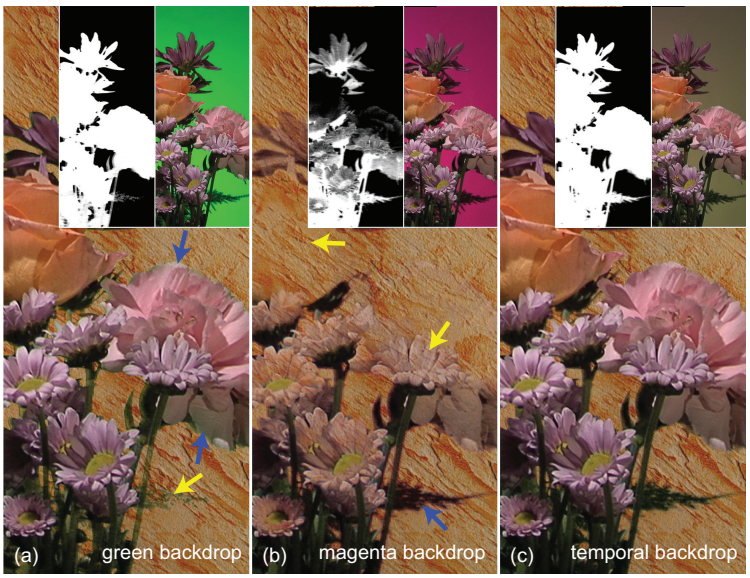


Figure 6.4: Another example of our proposed color spill neutralization. Note the completeness of the alpha matte as well as the neutralized color spill in (c). The errors of the individual alpha mattes as well as color spill is clearly visible in (a) and (b). Please zoom into individual image sections to see details of spill (blue arrows) and keying (yellow arrows).

C_b and the background colors B_a B_b:

$$s = \frac{|C_a - B_a| \cdot |C_a - B_b|}{|B_a B_b|} - \frac{|C_b - B_b| \cdot |C_b - B_a|}{|B_a B_b|} \tag{6.2}$$

Both B_a and B_b must be known in the camera's color space to normalize equation 6.2 with respect to the camera's response. They are determined by first applying the inverse of M_{maxb} to C_a and to C_b (to mask all background pixels) and then computing the two medians for the two sets.

If $|s|$ is above a threshold (T_s), then the corresponding pixel definitely belongs to a seam region. Since equation 6.2 is normalized, the threshold is constant for all frames. Empirically, we found that T_s=0.05 is suitable for a wide variety of recording scenarios. If s is negative, the seam color is caused by the background recorded in C_a, and C_b contains the correct foreground pixel. If s is positive, the seam color is caused by the background recorded in C_b, and C_a contains the correct foreground pixel. Otherwise, if $|s|$ is below T_s, then C_a and C_b contain foreground pixels.

All pixels classified as being located in a color seam region are marked in a binary error matte M_{err} (cf. figure 6.5h). To avoid misclassifications of moving highlights, regions with saturated intensity values in C_a or C_b are not taken into account. We apply a morphological opening to M_{err} to clean voids that are caused by camera noise.

Simply replacing the color seam pixels in C_{ab} by the correct foreground pixel from C_a or C_b will

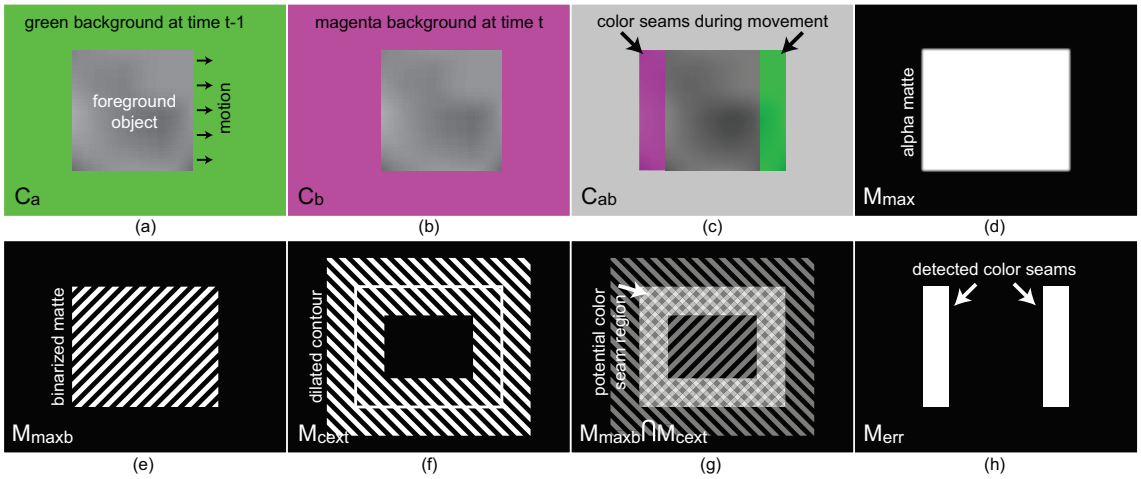


Figure 6.5: Steps of the color seam correction as explained in section 6.1.2: binarization (e), contour detection and dilation of contour (f), separation of color seam regions at the difference between M_{max} and M_{cext} (g and h). The further processing steps are described in figure 6.6

not neutralize their color spill. We correct this by processing only pixels that are marked in M_{err} as described in the following and illustrated in figure 6.6.

If a pixel in C_b was captured at time t and contains background, and the corresponding pixel in C_a was captured at $t+1$ and contains foreground (as illustrated in figure 6.6) then we compare the foreground pixel in C_a at $t+1$ with the corresponding pixel in C_b at $t+2$, using equation 6.2. If the latter is classified as foreground, both pixels are averaged to neutralize their color spill and the result is stored in C_{ab}. The same procedure is carried out with a pixel in C_a at $t-1$, if the corresponding pixel in C_b at t contains foreground and C_a at $t+1$ contains background. In this case, the pixel intensities in C_a at $t-1$ and C_b at t are averaged.

In the rare case of quickly moving thin foreground objects, this procedure is invalid, because a look back or a look ahead in the frame sequence will point to a background instead of foreground pixel. Again, this situation can be determined using equation 6.2. If this applies, we perform a spirally increasing spatial search around the original foreground pixel within the subset marked in M_{err} to find the closest neighbor that equation 6.2 identifies as a foreground pixel in both C_a and C_b. We do not simply average them as before to neutralize color spill, but try to preserve most of the information in the original pixel. Instead, we assume that the foreground color of the original (at location x,y) and the selected neighbor (at location i,j) may vary slightly, but their color spills are identical. We extract the color spill of the neighbor with

$$C_s(i,j) = \frac{C_a(i,j) - C_b(i,j)}{2} \tag{6.3}$$

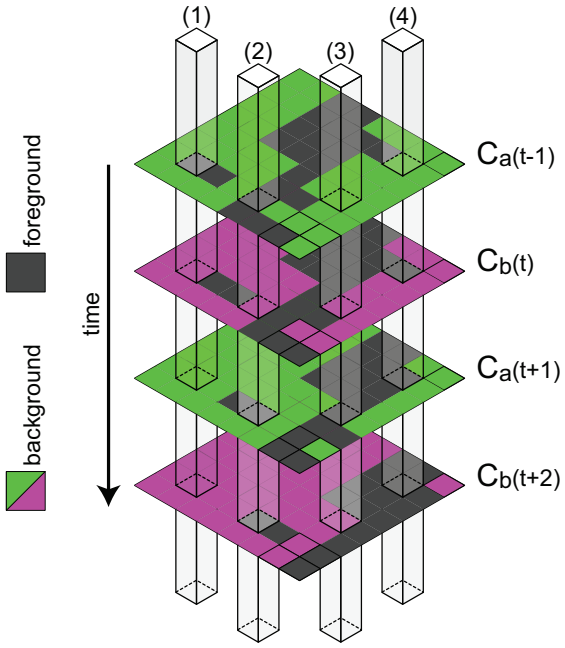


Figure 6.6: The four possibilities of pixel processing for times $t-1$ through $t+2$ are illustrated. For pixels 1 and 4, M_{err} does not contain an entry. Thus, their intensities are blended without any further processing. For pixel 2, $C_b(t)$ contains background information. Therefore, $C_b(t+2)$ is analyzed in a first step. In this example, it contains foreground information and can therefore be blended with $C_a(t+1)$. For pixel 3, however, $C_b(t)$ as well as $C_b(t+2)$ contain background. Thus, blending would not avoid the color seams in this case. Instead, the spatial neighborhood of pixel 3 in $C_b(t)$ and $C_a(t)$ is analyzed and the neighbor with the most similar intensity is used to compensate the color seam as described in 6.1.2.

if $C_a(x,y)$ contains the original foreground pixel, or with

$$C_s(i,j) = \frac{C_b(i,j) - C_a(i,j)}{2} \tag{6.4}$$

if $C_b(x,y)$ contains the original foreground pixel. The color spill of this pixel can then be suppressed by

$$C_{ab}(x,y) = C_a(x,y) - C_s(i,j) \tag{6.5}$$

or

$$C_{ab}(x,y) = C_b(x,y) - C_s(i,j) \tag{6.6}$$

depending on whether the original foreground was in C_a or in C_b.

Since the capturing rate is twice the final video rate, looking ahead by one captured frame leads to a constant delay of $1/2$ video frame. If this delay is not tolerable (e.g., for live broadcasting), then averaging with a corresponding foreground pixel in the look-ahead frame can be replaced by the spiral spatial search in the present frame, as explained above for quickly moving thin foreground objects. To compensate for the required larger number of search steps, the spatial search region can be reduced.

6.1.3 Temporal Backdrops

The required synchronized switching of the backdrop at a high speed can be realized in various ways: Video projectors with a low latency, such as DLP projectors, can be used in a front- or a rear-projection setup. To avoid temporal artifacts that result from a color wheel, 3-chip DLP projectors should be preferred over single chip projectors. Projector pairs with higher latencies are also applicable in combination with fast LCD shutters in front of their lenses. Applying projectors has the advantage that temporal backdrops can be generated on non-trivial scene geometry, as commonly used in studios. Additionally, low-latency flat panel displays can be employed. Commercial LCD, Plasma and FED panels support frame rates of up to 240 Hz, while the refresh rate of upcoming OLED and PLED displays might be even higher because of their significantly lower switching time of <1 μs. Temporal backdrops that are realized with a display technology can be coded temporally and spatially. The additional spatial coding can support camera tracking as outlined in section 6.4. However, if this option is not required, a simple temporal illumination of the background represents a cost-efficient alternative. This can be realized with RGB LEDs that illuminate white background diffusors, similar to [174], yet with alternating colors. Softboxes in the background that employ RGB LEDs for illumination is a further option. Another possible solution would be using reflective colored electronic ink displays. Although these devices currently suffer from very high latencies, faster devices may become available in the near future [177, 128]. In this case, the background has to be lit homogeneously, much like a standard blue screen.

No matter which approach is chosen, the backdrop has to be synchronized with the shutter of the camera and the complementary colors being used have to add up to white. Figure 6.7 shows several of our current implementations of temporal backdrops.

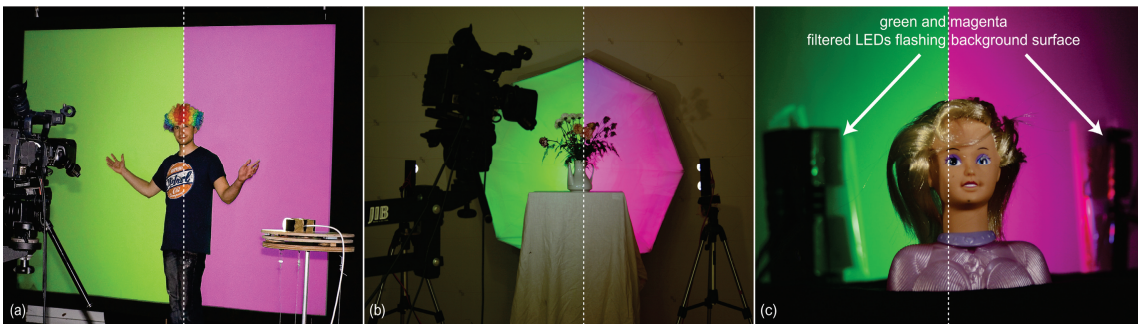


Figure 6.7: Different prototype implementations of temporal backdrops: a back projection screen (a), a softbox with integrated RGB LEDs (b), and a RGB front-illumination of a diffuse background surface (c). The foregrounds are lit with arbitrarily aligned spot lights.

6.2 Implementation

To synchronize the temporal backdrops with the shutter of the camera (a high definition 3CCD JVC GY-HD251 [25] in our case), we utilize the composite output signal of the camera as an external trigger. If the backdrop is realized with displays (e.g., projectors or flat panel screens), then we currently apply a Quadro FX 4500 X2 G-Sync whose G-sync input is connected with the composite output of the camera. The graphics card renders the images B_a and B_b synchronized to the camera's signal. To synchronize LED-driven backdrops, we use a custom-built electronic controller that is triggered by the VGA-out signal of the graphics card. All 720p frames are captured to disk via HD-SDI in *Yuv* 4:2:2 format at HD scanning speed of $59.94Hz$. Note that the compression of the chrominance channels leads to a slight reduction of the keying quality especially in the case of the magenta backdrops. A better quality could be achieved by using the uncompressed input data, but the limited write performance of the used hard disks of our prototype system did not allow the reliable recording of full 720p 8 Bit RGB data at $59.94Hz$ ($\sim 158MB/s$). Cameras which require Bayer patterns [7] or other spatially coded color filters [112, 101, 102] for color reconstruction, as well as cameras recording interlaced data require an interpolation of the color channels or scanlines respectively to ensure an equal resolution as well as correct registration of C_a and C_b. Because of the required precise synchronization of the camera and the temporal backdrops, camera systems with rolling shutters in which each pixel row is exposed and read out at a slightly later time than the previous row cannot be applied – even high-quality rolling shutter compensation algorithms which are applied in a post-processing step [20, 54, 5] cannot reconstruct the exact synchronization. A Camera with 3 CCD sensors and global shutter that records progressive images is applied in our implementation, since it provides the highest image quality.

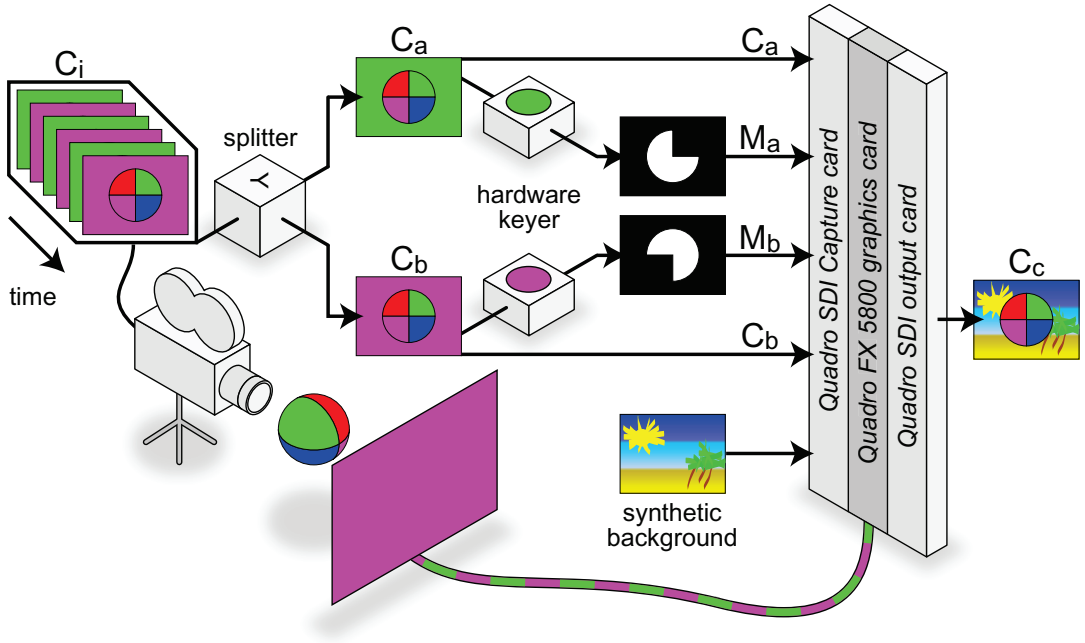


Figure 6.8: Studio-compatible hardware implementation of temporal backdrops: a splitter separates both backdrop states and forwards the corresponding images to hardware keying units. The computed alpha mattes and the original frames are forwarded to an Nvidia Quadro digital video pipeline that executes our color seam correction algorithm in real-time.

We chose green (G) and magenta (RB) as backdrop colors for the reason described in section 6.3, and we apply Ultimatte software chroma keying to compute M_a and M_b. All other processing steps explained in section 6.1.2 are implemented as Cg fragment shaders and are carried out in an average processing time of less than 17 ms (without texture up- and downloads) for high definition video resolution on an Nvidia Geforce 285 GTX. In our proof-of-concept implementation, we pre-record the video frames and apply software keying and compositing, since these steps are independent from our algorithm.

Figure 6.8 sketches a possible solution for integrating our technique easily into existing professional hardware composition pipelines to support real-time keying. The recorded video signal has to be split into C_a and C_b sequences using, for example, a Blackmagic Design Multibridge Pro module [131]. The two sequences can then be streamed to two hardware keying units, such as an Ultimatte 11 HD/SD [74]. Our algorithm can be executed at the pipeline's overall clock speed on an additional PC that is equipped with a genlocked Nvidia Quadro digital video pipeline system: The sequences of alpha mattes M_a and M_b, and the C_a and C_b sequences can be forwarded from the keying units to a Quadro SDI Capture card [117] for streaming them directly onto the GPU

memory of one or multiple Quadro FX 5800 graphics cards that are used for image processing and compositing. Additional Quadro graphics boards can be employed to drive all backdrop displays in sync with all other components. The final composite sequence can then be read out via a Quadro SDI Output card.

Alternatively, the entire pipeline (including the required algorithms for keying, seam correction, spill neutralization and compositing) can be implemented in on current state-of-the art GPUs, if compatibility to established professional software and hardware components is not required. By using GLSL or GPGPU languages like CUDA or OpenCL, the parallel processing power of current graphic devices offer sufficient performance to process 1080p content within the required frame-rates.

6.3 Results and Limitations

In figure 6.10, we compare our color spill neutralization to a uniformly illuminated white backdrop as ground truth. We also determine the color spill generated by single green and magenta backdrops, and evaluate how well it can be reduced by professional spill suppression algorithms (Ultimatte Advant Edge, library version 1.6.2 in our case). Therefore we transformed the composite images from *RGB* into the $CIEL^*a^*b^*$ color space and calculated the perceived chrominance difference by applying the ΔE^*_{ab} color difference formula [137] to the two chrominance channels:

$$\Delta E^*_{ab} = \sqrt{\Delta a^{*2} + \Delta b^{*2}} \qquad (6.7)$$

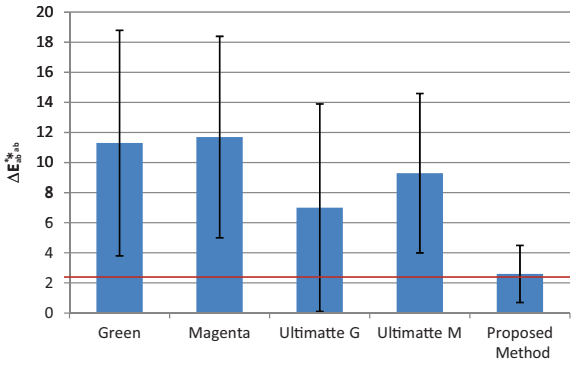


Figure 6.9: The bars visualize the average ΔE^*_{ab} as well as the standard deviation for the images shown in figure 6.10. Our proposed method clearly outperforms the other solutions and reduces the visibility down to the JND point (red bar).

Table 6.1 provides the measured ΔE^*_{ab} chrominance differences to the ground truth for the samples shown in figure 6.10. Our method clearly outperforms professional spill suppression techniques. This is also visualized in the diagram 6.9.

While, on average, such techniques reduced the ΔE^*_{ab} of visible color spill only by a factor of 1.4 in our experiments, our technique compensated its ΔE^*_{ab} by a factor of approximately 4.4 – down to $\Delta E^*_{ab} = 2.6$. Note that a ΔE^*_{ab} of 2.3 represents the visibility threshold and equals one just no-

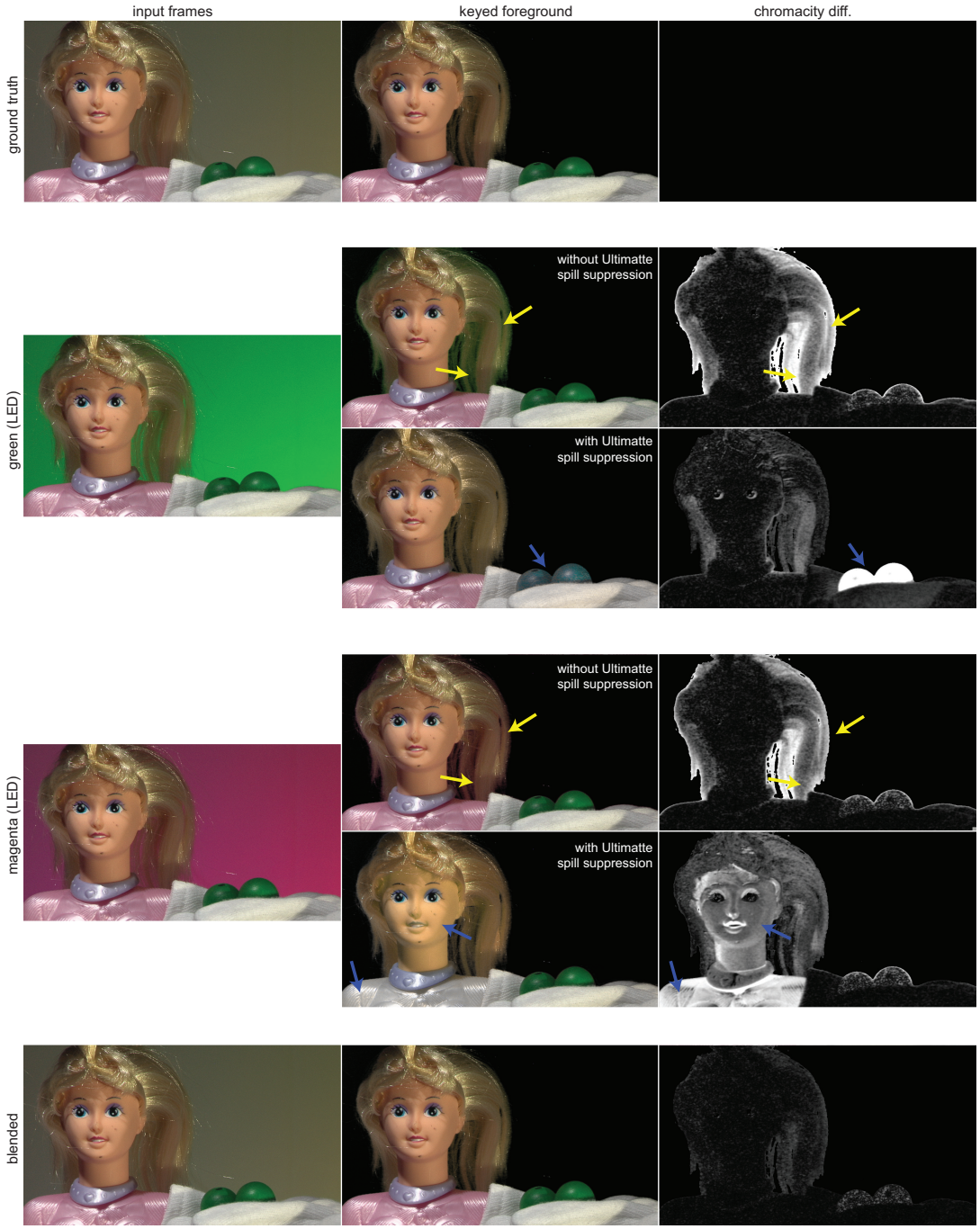


Figure 6.10: Comparison of color spill neutralization and spill suppression: a scene with a uniformly lit white background serves as ground truth (upper row). The remaining color spill is calculated from the ΔE^*_{ab} chrominance distance in CIE L*a*b* color space – with and without software spill suppression (two center rows for green and magenta backgrounds). The result of our color spill neutralization is shown in the bottom row. The yellow arrows indicate errors caused by remaining color spill, while the blue arrows point to wrong color reproductions caused by spill suppression.

ticeable difference (JND) [60]. Note that the software spill suppression used in Ultimatte fails for foreground pixels with color values similar to the backdrop, as shown in figure 6.10. This is not the case when our approach is applied. As explained above, we transform color spill into a neutral (white) background illumination by averaging C_a and C_b. It is also imaginable to entirely remove spill by subtracting the absolute values of $C_a - C_b$ from C_{ab}. This would mimic a neutral black background, but is only possible for entirely static setups, since non-registered pixel intensities with different colors will lead to a false color reproduction.

Input	**Average** ΔE^*_{ab}	**Maximum** ΔE^*_{ab}	**Standard Deviation**
Green	11.3	26.7	7.5
Magenta	11.7	27.7	6.7
Green Ultimatte	7.0	32.8	6.9
Magenta Ultimatte	9.3	50.4	5.3
Proposed Method	2.6	5.0	1.9

Table 6.1: ΔE^*_{ab} chromacity differences for the sample shown in figure 6.10.

It was shown in [68] as well as in [174] that applying motion interpolation for computing intermediate frames is efficient for time-multiplexed techniques with no real-time demands. If no real-time processing is required, motion interpolation can also be applied to reduce the color seam regions of the proposed approach before its correction.

Additionally, the presented temporal chroma keying can be combined with a natural feature matting, such as Bayesian image matting [36] to further enhance its quality. This can be fully automated by applying an eroded and a dilated version of M_{maxb} to compute a trimap, as illustrated in figure 6.11. Note that also in this case the matting has to be applied independently to C_a and C_b and the results have to be combined by the maximum operator.

In all our prototype setups, the backdrops are temporally switched at 59.94 Hz. Although this is above the critical flicker frequency of the human visual system [110], slight alternating backdrop intensities can be perceived. This is reduced to a minimum by choosing green and magenta as backdrop colors, since their integrated luminance can be matched easily as explained in [191] which reduces the perceptibility of the consecutive chrominance changes [98]. The remaining brightness alternations are barely detectable. Color seams are observable at the edges of the backdrops during fast eye movement. Both, however, did not significantly distract people when working with temporal backdrops. By using high speed cameras and illumination devices, this issue can be reduced even further.

The irradiance of the backdrop is mixed with the illumination of the foreground. This leads to a reduced color saturation that depends on the amount of light arriving at the backdrop. In our experiments, however, we measured a foreground illumination arriving at the backdrop that was more than twice the actual backdrop brightness[28] but did not affect the quality of the resulting alpha mattes. Nevertheless, our method has limitations: One fundamental drawback of chroma keying is that

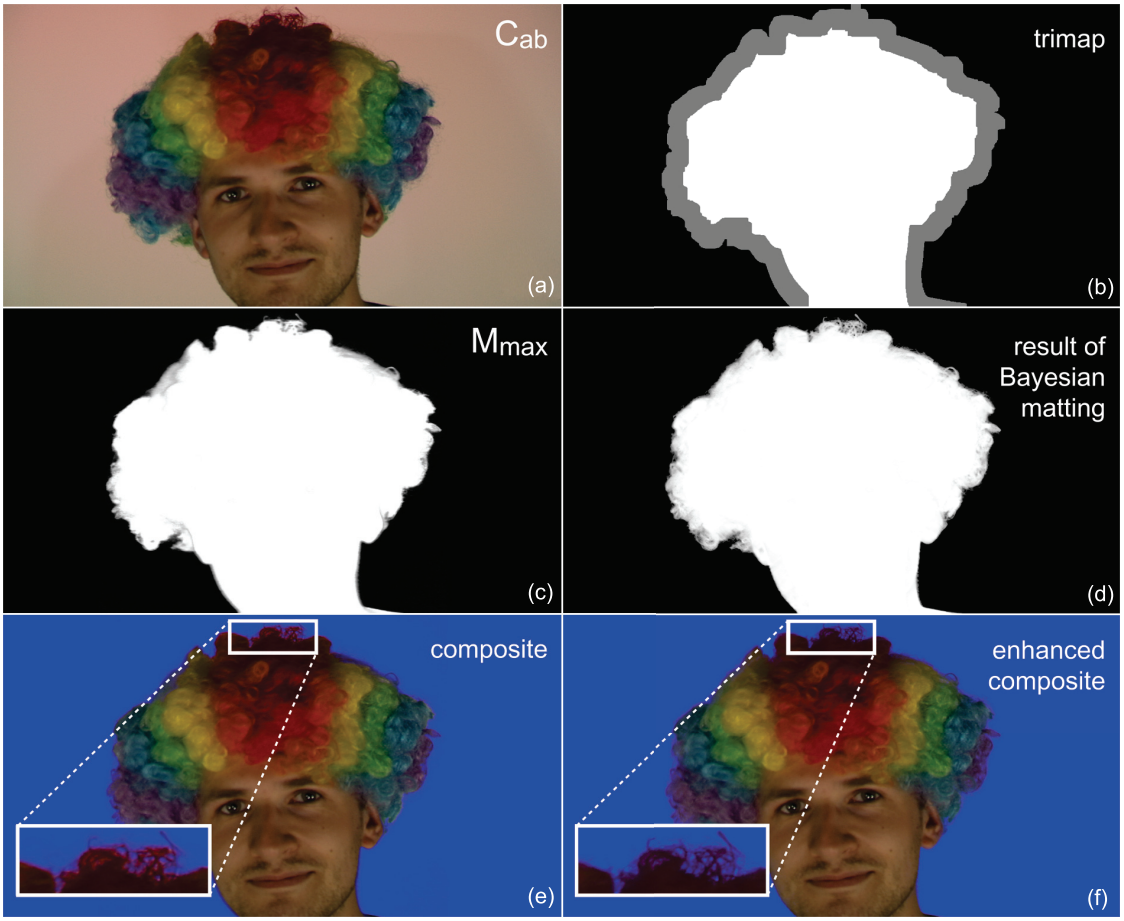


Figure 6.11: Keying enhancements through additional Bayesian matting for applications without real-time demands: A trimap (b) can be computed automatically by eroding and dilating M_{maxb}. After applying Bayesian matting to C_a and C_b, the maxima of both results are used to calculate improved alpha mattes (d and f) as does chroma keying alone (c and e). Please zoom into individual image sections to see details. Note the reduced white seam resulting from wrong alpha values in the close-up.

strongly reflecting or refracting foreground objects are not keyed correctly. Furthermore, in the rare case that moving foreground objects contain neighboring complementary colors that are identical

[28] 110 Lux vs. 48 Lux

to our backdrop colors, keying also fails. Strong motion blur resulting from rapid movements can lead to lower keying quality. This, however, is also a common problem for standard chroma keying with static backdrops. High-quality motion interpolation and shorter shutter times will reduce this effect.

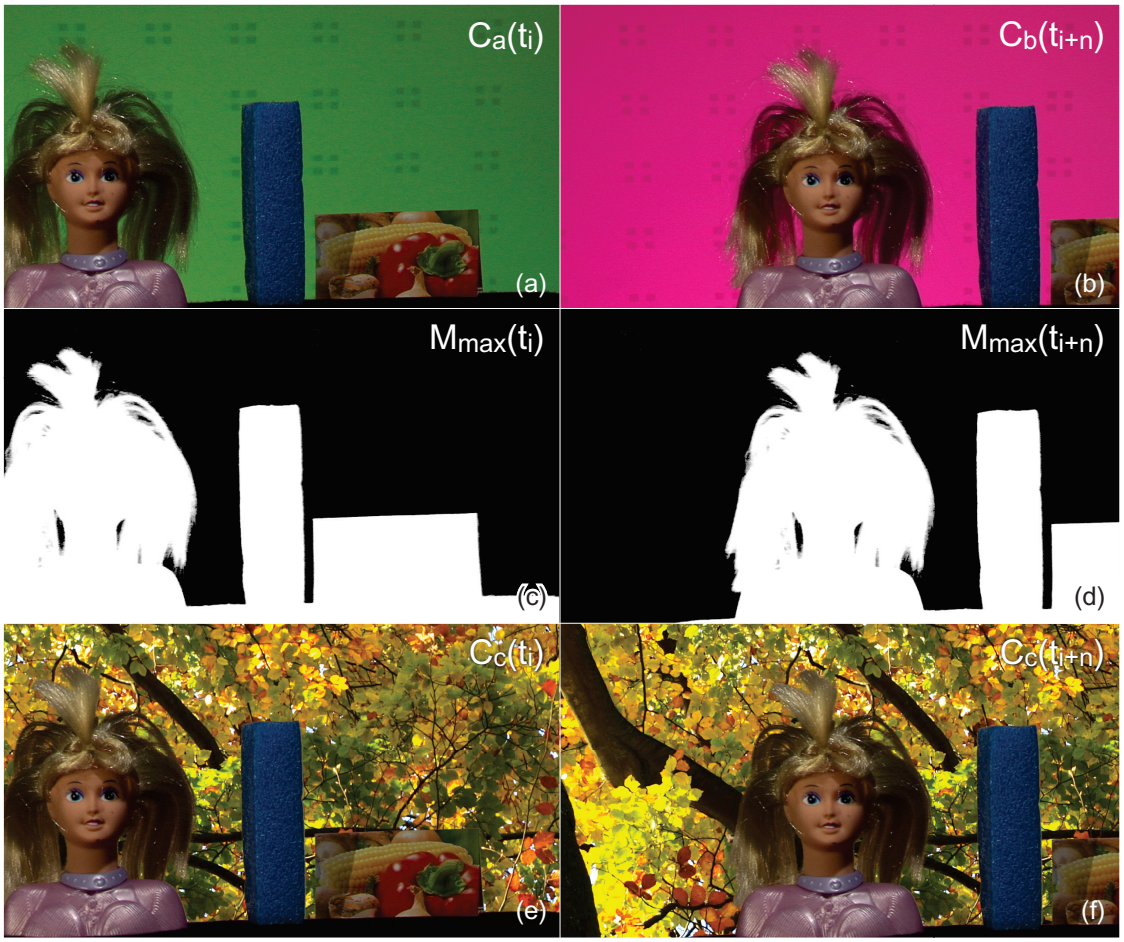


Figure 6.12: Supporting camera motion tracking through embedded spatial codes: frames C_a at time t_i (a) and C_b at time t_{i+n} (b) with embedded feature points. Reducing the luminance of the displayed code features does not influence the quality of the computed alpha mattes (c and d). The final composites (e and f) display the result of a perspectively augmented synthetic background as a result of motion tracking.

While geometrically complex studio props can, in principle, be transformed into temporal backdrops using front- or back-projection, they are easier to manufacture from uniformly colored material that can be keyed. Such physical props are often used to give actors or moderators orientation with respect to virtual props. This, however, can alternatively be achieved with a synchronized

projection, as explained in [68]. For outdoor sets, static backdrops might still be more efficient than temporal backdrops.

6.4 Summary and Discussion

We have presented a novel matting approach that improves conventional chroma keying in two ways: firstly, our color spill neutralization method leads to significantly better color reproduction than digital color spill suppression techniques, is fully automatic and supports real-time rates. Secondly, the temporally switched complementary colors make chroma keying invariant to any foreground color, and a uniform and consistent background illumination is easier to achieve with emissive or rear-illuminated backdrops than with reflective ones that are front-illuminated. It allows an ad-hoc adjustments of scene lighting while remaining robust keying results.

In contrast to related approaches that are alternative to standard chroma keying, our method is not variant to distinct optical modulation of foreground materials, such as polarization, retro-refection and infrared reflection, supports dynamic foregrounds and camera motion. It is fully compatible with conventional hardware composition pipelines which are used in professional studios, and can be combined with motion tracking techniques. The latter is demonstrated in figure 6.12, where a back-projection display is used as temporal backdrop for spatially encoding static feature points in addition to a temporal color coding for supporting simultaneous camera motion tracking. The feature points are displayed during both backdrop states and have the same chrominance as the corresponding backdrop color, but are slightly reduced in luminance. As explained in [68], they can then be used as input for classical match-moving algorithms that reconstruct the 3D camera path or for motion-tracking techniques that determine the 2D displacement of the camera with respect to the background plane. In our experiments, a luminance reduction of 15% was sufficient for robust feature detection without affecting the matting process.

7 HARDWARE SETUP

Besides the algorithm development and software implementation described in the last chapters, specialized hardware modules were implemented for all the developed systems. These were required as control units for the correct frame-precise synchronization of the camera, the LEDs and projector illumination. While the setup varied from case to case, the main parts were consistent in all the developed system. The components as well as their connections are summarized in figure 7.1.

The following sections describe the different hardware implementations in detail. First an overview of the illumination control (i.e. LED unit and projection) is given, followed by the different types of synchronizations used and finally the steps used to calibrate the respective cameras as well the projectors.

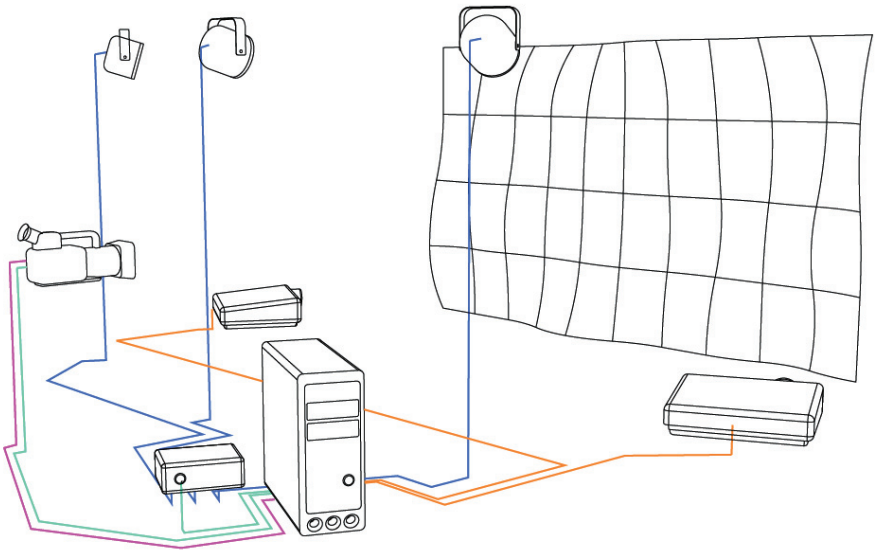


Figure 7.1: Schematic diagram of the principal hardware components and their connections: the projectors receive the already synchronized signal directly from the graphics card (orange lines). The camera transfers the recorded image data to the computers main memory either via a frame grabbing device or via Firewire 800 (magenta line). Additionally, it either generates the synchronization signal and sends it to the graphics board or it receives it from the GPU's v-sync output (cyan line). This signal is also used to synchronize the illumination control unit that controls the attached LEDs (blue lines).

7.1 Illumination

Applying temporal and spatial illumination modulation at a recording frame rate between $60 - 120$ Hz requires low latency components. For this reason, analog light sources, such as standard light bulbs or halogen illumination could not be used as their fade in and fade out take several milliseconds and are, therefore, much too long. Furthermore, the $50 - 60$ Hz frequency of standard alternating current leads to temporal flickering which makes it impossible to generate homogeneous illumination even if the intensity of the light sources are not changed over time. To avoid these problems, a custom-built LED illumination system driven by direct current was used which is capable of a high frequency switching as well as illumination adjustment via pulse width modulation (PWM).

Projector-based illumination, similarly, also requires precise synchronization which restricted the application to DLP projectors or LCD projection devices with custom build LC-shutters mounted in front of the optics.

7.1.1 LED Controller

To synchronize the LEDs precisely to every recorded frame, a custom micro controller based system was developed. This system uses the v-sync signal delivered by the analog video output of the graphics device as input [29]. It consists of a USB-powered electronics unit that converts the analog blanking signal of the v-sync output into a rectangular pulse suitable for synchronization.

The output signal of the synchronization unit was devised as input for another custom-made micro controller unit which was devised to control the illumination strength of the individual LEDs by PWM. Please refer to appendix A.1, where figure 1.2(a) shows a circuit diagram of the control unit and the board layout is shown in figure 1.2(b). Note that this diagram shows the general setup which was slightly modified depending on the amount of LEDs used and how they should be controlled. The BASIC source code can be found in appendix A.2.

[29] cf. figure 1.1(a) and 1.1(b) in appendix A.1 for a diagram of the custom electronics unit used to convert the v-sync signal into the required trigger pulse.

7.1.2 Projection

To generate frame-precise image content with the projectors, the quad buffer mode of an Nvidia Quadro graphics device was used. While this is normally used to guarantee continuous switching between the right and left eye view for active stereoscopic rendering, here it was utilized to guarantee alternately changing image content in every other frame. Depending on the required synchronization mode, which in turn depends on the camera used, a synchronizable graphics card was required. For this reason we used an Nvidia Quadro FX4500X2 G-Sync [119] which supports the use of external synchronization sources, as described in the next section.

7.2 Synchronization

For the on-line as well as off-line processing it is fundamental that all components are perfectly synchronized. This means that the LED lighting system, as well as the projectors, must be synchronized to the camera's frame rate and shutter operations. Depending on the hardware used this can be achieved very precisely or with less precision leading to constraints which, however, had little influence on the quality of the setups.

7.2.1 On-line Processing

For the 120 Hz real-time setup required for the adaptive imperceptible coded projection described in chapter 4, DepthQ Stereo DLP projectors [46] were used which are capable of projecting 120 frames per second at SVGA [30] resolution. A PTGrey Dragonfly Express camera [152] was used for recording 120 frames at VGA [31] resolution. The camera offers an external trigger input which was used to receive the v-sync signal of the analog video signal of the projectors as a trigger signal for image exposure (cf. section 7.1.1 and figures 1.1(a) for a schematic circuit diagram and 1.1(b) for the board layout used. Both can be found in the appendix A.1). Instead of using the theoretical maximum of $1/120$ s $= 8.\overline{333}$ ms as per-frame exposure time, this could only be set to a maximum value of $\sim$7 ms due to the fact that the camera requires approximately $\sim$1.3 ms for its CCD sensor readout. This did not degrade the quality of the recoded footage because the used DLP projectors contain a 4-segment color wheel with red, green, blue and a final white section. The final white color segment was deactivated within the setup, because otherwise the internal color processing

[30] 800x600 Pixels
[31] 640x480 Pixels

unit of the projector would have modulated the input intensities to enhance the perceived contrast. This, however, would have lead to uncontrollable non-linear behavior of the positive and negative Δ-values which would have made the embedded code perceivable. Thus only $\sim$6.25 ms are required to record the complete red, green and blue intensities of the projected illumination.
To robustly process the complete data at a rate of $37MB/s$ [32], an IEEE1394b (Firewire 800) connection was used and the required demosaicing of the Bayer filter was accelerated by a GPU-based implementation. The LED lighting system was also synchronized by the v-sync signal of the VGA output, as described in section 7.1.1, and a custom-built controlling unit was used to trigger and dim the LEDs required for illumination and flash keying via PWM.

7.2.2 Off-line Post-Processing Applications

To achieve higher image quality for production use, a JVC GY-251E three chip HD studio camera [25] was used for image recording in the off-line systems introduced in chapters 5 and 6. The raw image data was streamed via Serial Digital Interface (SDI) to a AJA Xena HD framegrabber device [2] which compressed the signal from RGB into Yuv 4:2:2 color space to enable sustained real-time recording at a rate of $\sim$103 MB/s on the standard consumer hardware used. The camera offers an analog composite signal output which was used as a synchronization signal for the complete system. This signal was transferred into the genlock input of a Nvidia Quadro FX4500X2 G-Sync [119] graphics board which allowed to use it to synchronize the used projectors to the camera's fixed frame rate of 59.94 Hz. In this setup the camera's shutter time had to be reduced to a maximum value of 12 ms, because it required approximately 4.5 ms to read out the three CCDs and to transfer the data via SDI to the framegrabber card. Where DLP projectors were used in the system, care had to be taken to use projectors with at least one additional fourth white color wheel segment, like the projectors described in the previous section. By deactivating this, only approximately 0.5 ms of the third (blue) color channel was lost from the captured imagery, although this had little influence on the quality of the reconstructed codes.

The illumination, likewise, was triggered via the VGA v-sync signal of the pre-synchronized projectors as explained in section 7.1.1. Depending on the configuration of the LED lighting system, the hardware controlling unit was dynamically re-programmed via a USB interface to support flashed illumination (cf. chapter 5) or time sequential green-magenta LED illumination (cf. chapter 6).

[32] 640x480x120 Bytes of image data

7.3 Camera and Projector Color Response Calibration

The camera's non-linearities were neutralized by measuring the the response curves in all three color channels. This was carried out once in advance by capturing an images series of a static scene with increasing shutter times and estimating the camera's intensity transfer function from that information as described in [43] to recover the response curves for all three color channels. Before processing, the recorded camera images were linearized by a pre-calculated lookup-table operation. While a linearized camera image is not necessarily required for the matting process of the different applications, it is crucial during acquisition of the surface's reflectance properties required in the system described in chapter 5 for radiometrically compensated projection, as well as the precise recovery of the embedded imperceptible code patterns as described in chapter 4.

The individual projector responses were linearized by a Datavision Spyder2Pro calibration device [153]. To guarantee consistent radiometric compensation results in the case of overlapping projectors, all projector's intensities were re-mapped to the range of the one with the lowest overall luminance.

8 Conclusion and Outlook

8.1 Conclusion

It is hard to imagine the generation of high-quality special effects without the profitable film and television industry, and vice versa. Increasing processing power, new controllable illumination units such as projectors with high spatial and temporal resolution for illumination adaptation, as well as LEDs which can be modulated even much faster have opened up new possibilities for video compositing in terms of enhanced keying, environment matting as well as imperceptible pattern projection for various applications. In addition, projectors can be used to generate these effects at arbitrary locations without the need for special virtual studio sets. Furthermore, they can be used to display dynamic direction hints for the actors, which are not visible in the final composites.

In this dissertation, we have introduced a set of novel methods of digital video compositing using synchronized projector, camera and lighting systems. By modulating the illumination spatially as well as temporally, a variety of codes can be embedded to enhance different kinds of applications and to broaden the possibilities of traditional digital video compositing. In summary, the following achievements were presented in this thesis:

- The evaluation, implementation and verification of a real-time adaptive imperceptible code projection algorithm
- The application of the imperceptible code projection developed in the context of a real-time compositing system enabling optical camera tracking with invisible markers and alpha matting by means of flash keying
- The development of a synchronized LED-projector-camera system capable of the generation of various digital video compositing effects outside of completely controllable virtual studios, i.e. in arbitrary film settings
- The realization of a temporally modulated chroma keying method which uses alternating complementary colors to overcome the limitations of the classical method in terms of color dependency as well as color spill
- The development of specially designed control units to drive and synchronize the LED units, the projectors as well as the cameras

In the remainder of this section we will summarize these achievements in more detail.

We presented an innovative imperceptible embedded code projection technique that, in contrast to previous work, takes into account various parameters of human visual perception for the optimal encoding and decoding of integrated patterns during normal viewing, but also during saccadic eye movements as well as code transitions. It is based on high-frequency temporal image modulation. Furthermore, it can be applied with unmodified projectors and does not require advanced calibration other than linearization and a gamma correction. In combination with the described imperceptible code projection we demonstrated a real-time flash keying approach for alpha matting. By combining the developed techniques in a proof-of-concept prototype of an Augmented Reality-TV studio, a dynamic multi-resolution marker method was introduced enabling robust in-shot optical camera tracking.

We demonstrated the prototype system of a compact *Virtual Studio 2 Go*, which is a concept offering all the flexibility of a virtual studio at film sets with arbitrarily shaped and textured diffuse surfaces as backgrounds. Using a synchronized projector-camera system with temporally multiplexed LED illumination enables the geometrical and radiometrical undistorted projection of keying patterns and other spatial codes onto the surface. This is carried out by using initially acquired calibration data for geometric correction and radiometric compensation. While these codes can be used to generate alpha mattes similar to the method used in a virtual studio, they can further be used for the generation of environment mattes, for camera path reconstruction or to project dynamic hints for moderators directly on the surfaces. The flashing LEDs record the fully lit scene at every other frame. Thus, after reconstructing the intermediate frames to generate two synchronized video sequences, every pixel of the video stream containing the LED illumination can be used for the final composite without having any physical blue screen blocking parts of the camera's view.

The temporal chroma keying approach using two complementary colors which are displayed alternately at the recording rate is able to overcome the two main limitations of classical chroma keying methods: its color dependency by calculating two individual alpha mattes for the complementary colors and combining both by a maximum operation, and the problem of color spill by blending two consecutive images with each other which transforms the color spill into neutral white backdrop illumination. The edge artifacts resulting from temporal misregistrations between two consecutive frames could be efficiently handled in an automated post-processing step. While this was implemented off-line in our prototype system, its GPU-accelerated implementation supports real-time rates and can, therefore, be integrated in a professional live broadcasting system.

The various components of the different systems were synchronized and partially driven by custom-built micro controller synchronization units. Depending on the camera used, either its internal composite signal was used as a synchronization trigger signal, if available, or the graphics card's v-sync signal was used as a reference clock for the other components. While the custom-built control units were implemented only as prototypes, it is conceivable that freely configurable LED control units that can be synchronized to external signals will become available in the near future.

8.2 Outlook

Some particular aspects of future work have been already discussed individually for each of the three presented synchronized camera-illumination systems in the corresponding chapters. In the final part of this thesis, however, we will offer a broader outlook on future research directions which could not be carried out within the context of this dissertation due to time as well as financial limitations.

The adaptation parameters used to calculate the maximum possible Δ-value for the imperceptible coding scheme described in chapter 4 were derived from an evaluation with a few subjects and only for one (the only available) hardware setup. While the results proved to be able to generate acceptable results with the framework used, a much more thorough user study would have to be carried out to generate a universally valid formula for arbitrary setups with varying illumination strengths and different frame rates. Furthermore, the influence of the individual color channels on the noticeability of the embedded codes as well as the perceptibility of other, non-binary code patterns could be evaluated further. In our system, we heuristically reduced the green color channel by a fixed value that seemed to be reasonable, but a more extensive evaluation may have the potential to find out optimal parameters for all individual color channels. Besides the application of imperceptible coded projection for digital video compositing or 3D reconstruction, it could also be imagined to use this technique for other purposes, such as digital watermarking for cinemas: while the embedded codes are not visible to human observers, unsynchronized cameras will record distorted images depending on the temporally varying embedded code patterns.

The adaptive, dynamic marker placement described in chapter 4 offers the possibility to guarantee an optimized visibility of markers for the current camera pose. The presented technique to adapt the marker size depending on the visibility of different regions of the projection plane could also be further extended to dynamically place markers at suitable locations on arbitrarily shaped projection

surfaces. As explained in chapter 5, its geometry could be reconstructed and analyzed for planar subregions to dynamically adapt the marker positions and sizes depending on the current camera position as well as possible occlusions.

While the generation of digital video composition effects in arbitrary locations using the synchronized illumination modulation presented in chapter 5 showed great potential as an off-line post-processing solution, it could also be implemented as a real-time system which would make it even more advantageous. While we decided to generate the effects as post-processing steps because of limitations in processing power and visual quality of the composites, new generations of GPUs offer enormous parallel processing power and specialized high-level general purpose programmable languages such as CUDA [118] or OpenCL [88] which would make it possible to apply the required frame-wise interpolation in real-time by using algorithms similar to that presented in [86]. In addition, it could conceivably be possible to generate a medium-quality GPU-accelerated real-time optical camera tracking solution as, for example, the solutions proposed in [187] and [89]. On the other hand, the system could also be further extended for dynamic, but completely controllable situations, where no real-time recording is required, such as stop-motion productions. It could be imagined to generate a high resolution geometrical scan of the current frame and generate high-quality environment mattes and light transport scans to generate projector based lighting simulations or photorealisitc off-line re-illuminations.

Besides alpha matting, environment matting, optical camera tracking as well as visual hints for moderators, the coded high speed illumination approach also has the potential to be applied in the context of re-illumination to generate relighting effects similar to Bayesian Relighting [56] or the impressive techniques proposed by Debevec et. al. [44, 82, 174]. While initial experiments into applying re-illumination via three light source photometric stereo generated severely noisy, low quality results using the available hardware components, it definitely has the potential to generate high-quality results similar to the ones presented in [174] if more powerful illumination systems and faster synchronization are used.

In addition to the technical innovations, the proposed systems also require new kinds of interactive authoring tools especially for their application within real-time live-compositing systems. Adapting to the actor's situations in real-time can make it necessary to directly adjust, for example, virtual components of the composite (which would be similar to a standard virtual studio production) as well as affect the generation and adaptation of projected moderator hints.

Besides professional visual effects production, we envision applications for the temporal backdrops proposed in chapter 6 in other domains such as novel game interfaces. For instance, video cameras such as Sony's EyeToy or Microsoft's Xbox Live Vision are becoming ever more popular components of gaming consoles to integrate players directly into games or movies [79]. A simple LED front illumination that temporally flashes the walls behind the player (similar to that illustrated in figure 6.7c in section 6.1.3 at a smaller scale) might help to make the keying process more robust. Visually, this would be perceived as plain white illumination.
We presented prototypes with active illumination based on projectors or LEDs. In the long term, one could also imagine the use of other materials, such as, the walls of a novel virtual studio seamlessly equipped with synchronized organic light emmiting diodes (OLEDs) or novel kinds of low-latency electronic ink (E-Ink) serving as passive, adjustable backdrops.

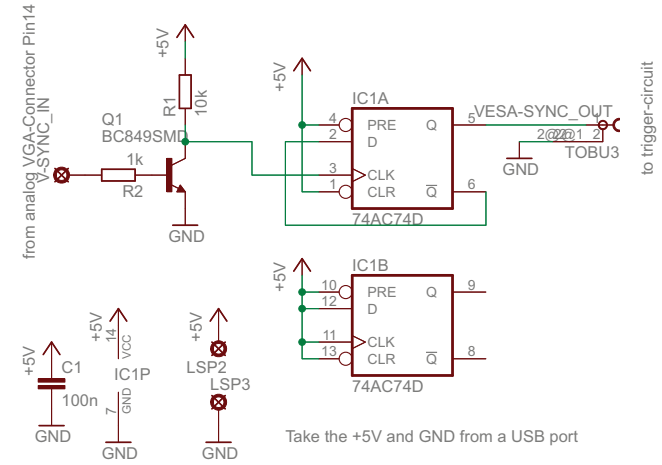
Using a synchronized ultra high-speed system working at frequencies above 1000 Hz would make it possible to further extend the possibilities of such systems by using imperceptible structured light projections for 3D reconstruction, the generation of enhanced environment mattes and relighting. While this can be realized technically using specialized hardware components like the Phantom camera series [165], custom-built projectors [77] and a high-speed bus and memory system, the complete system would become unaffordable for a majority of users. If, however, cost is not an issue, such a system would be able to generate a series of professional digital video composition effects which cannot be achieved with standard tools.

In this dissertation, new possibilities for synchronized, spatially and temporally modulated illumination for video compositing effects were explored. We believe that our presented work as well as the possibilities listed as future research directions may encourage others to further investigate the wide-ranging capabilities of such kinds of systems to broaden their application in the field of digital video compositing.

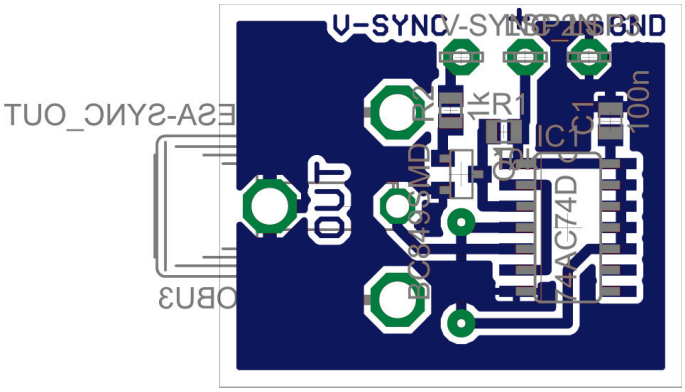
A APPENDIX

A.1 Circuit Diagrams and Board Layouts

The following pages display the circuit diagrams as well as the board layouts for the v-sync to trigger signal conversion unit, the LED controller and the connection required to externally trigger the Dragonfly Express camera.

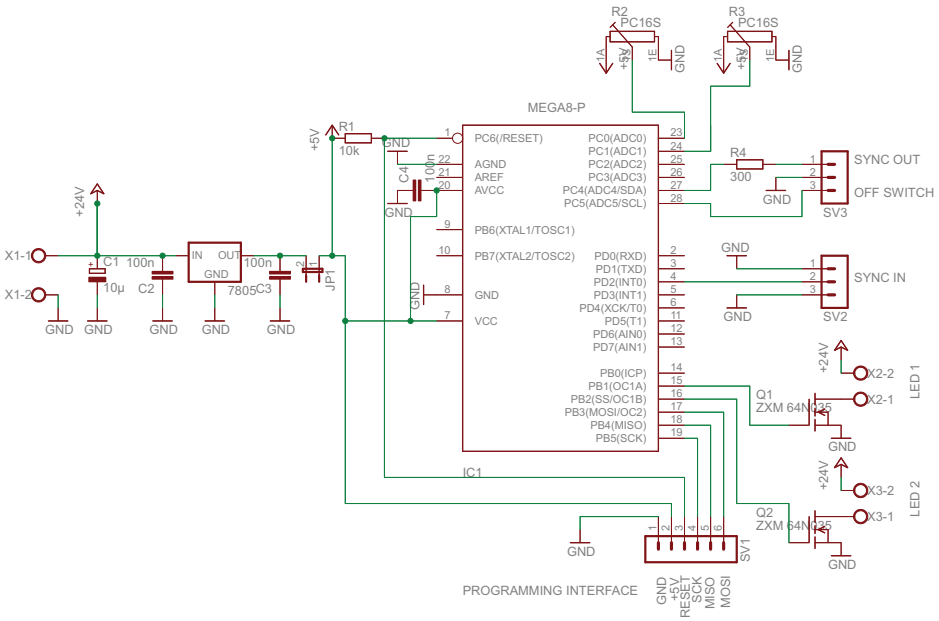


(a)

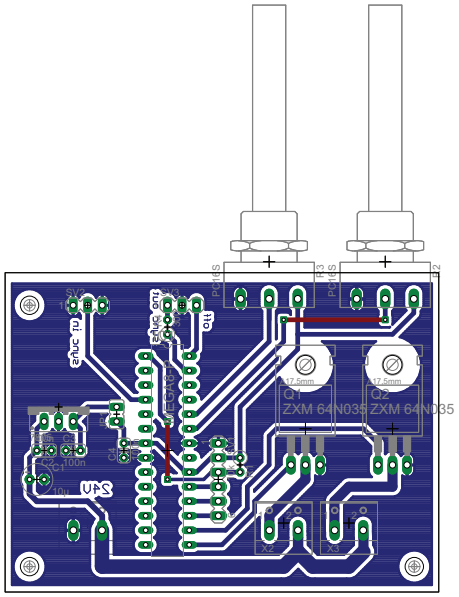


(b)

Figure 1.1: Circuit diagram (a) and board layout (b) of the v-sync to trigger signal converter.

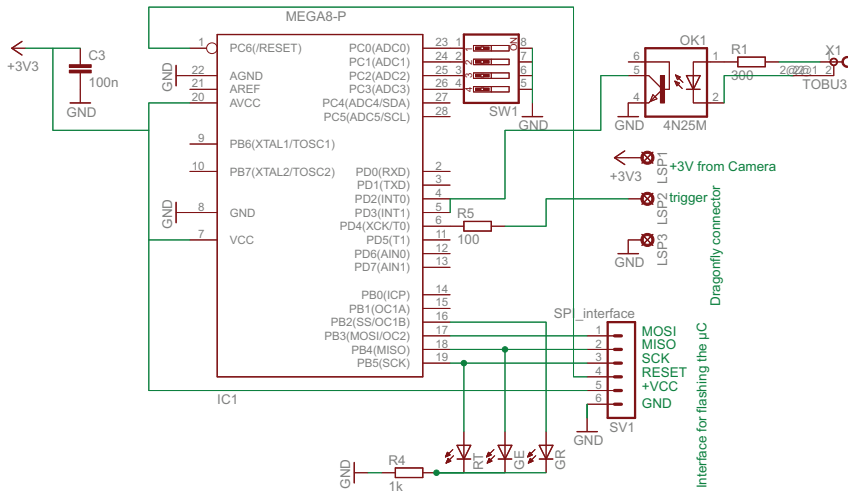


(a)

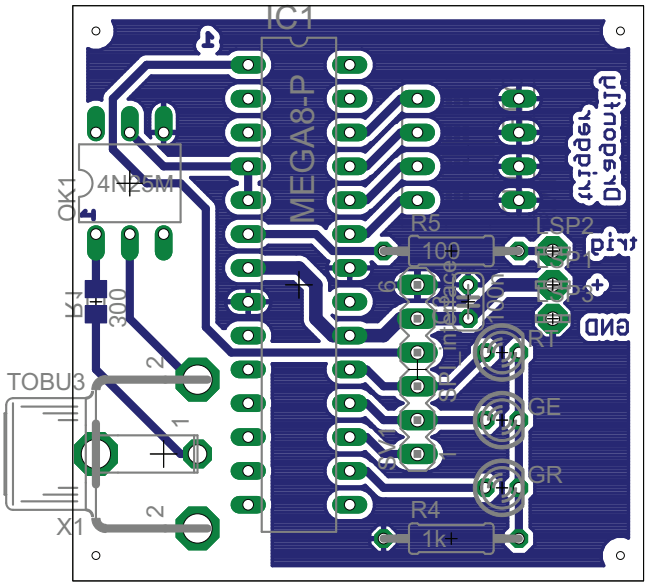


(b)

Figure 1.2: Circuit diagram (a) and board layout (b) of the two channel PWM-controlled LED synchronization and dimming unit.



(a)



(b)

Figure 1.3: Circuit diagram (a) and board layout (b) of the implemented camera trigger unit.

A.2 LED controller source code

The following BASIC source code was used to synchronize and control the LED units used for the described prototype systems. The BASCOM AVR BASIC compiler [38] is required for compiling. Details about the required hardware are given in chapter 7.

```
1  $regfile = "m8def.dat"
2  $crystal = 8000000
3
4  Config Portc = Input     'ADC pins, pinc5 = off switch (connects pin C5 to GND)
5  Config Pinc.4 = Output   'pin C4 = vesa_sync out
6  Portc = 0
7  Set Portc.5              'pullup pin C5 (off switch)
8  Config Pinb.1 = Output   'to MOSFET for OSRAM led
9  Config Pinb.2 = Output   'to MOSFET for OSRAM led
10 Config Pind.2 = Input    'vesa_sync in
11 Set Portd.2              'sync pin can be pulled low by transistor
12
13 Compare1a = 0
14 Compare1b = 0
15 Config Timer0 = Timer, Prescale = 1024
16 Config Timer1 = Pwm, Pwm = 10, Compare A Pwm = Clear Down, Compare B Pwm = Clear Down, Clear Timer = 1, Prescale = 1
17 Config Adc = Single, Prescaler = Auto, Reference = Avcc
18
19 Config Int0 = Rising
20 On Int0 Light
21
22 Declare Sub Turn_off
23 Declare Sub ShowTest
24
25 Dim Fade As Single , X As Single , Lowled1 As Word , Lowled2 As Word , Led1 As Word , Led2 As Word
26 Start Adc
27 Enable Int0
28 ShowTest                 ' call LED test sequence
29 Enable Interrupts
30 Timer0 = 0
31
32 'mainloop
33 'divide brightness by 2 if there is no
34 'sync-signal (to have same resulting brightness on LEDs)
35 Do
36    If Timer0 > 250 Then
37          Stop Timer0
38          Lowled1 = 1023 - Getadc(0)
39          Lowled2 = 1023 - Getadc(1)
40          Shift Lowled1 , Right , 1
41          Shift Lowled2 , Right , 1
42          Compare1a = Lowled2
43          Compare1b = Lowled1
44    End If
45    Turn_off
46 Loop
47 End
48
49 'turn-off-sequence
50 'check state of PIN C5 and
51 'connect/disconnect PWM from port pins
52 Sub Turn_off
53    If Pinc.5 = 0 Then
```

```
54        Config Timer1 = Pwm, Pwm = 10, Compare A Pwm = Disconnect, Compare B Pwm = Disconnect, Clear Timer = 1, Prescale
              = 1
55     Else
56        Config Timer1 = Pwm, Pwm = 10, Compare A Pwm = Clear Down, Compare B Pwm = Clear Down, Clear Timer = 1, Prescale
              = 1
57     End If
58  End Sub
59
60  'testsequence after switching on the circuit
61  'fades on and off both LED-channels sequentially
62  Sub ShowTest
63     X = 0
64     Waitms 500
65     Do
66       Fade = 1 - Cos(x)
67       Fade = 256 * Fade
68       Led1 = Abs(fade)
69       Led2 = 512 - Led1
70       If X > 3.14 Then Compare1b = Led2
71       If X > 6.26 Then Compare1a = 0 Else Compare1a = Led1
72       Waitms 1
73       X = X + 0.005
74     Loop Until X > 9.42
75     Wait 1
76     Led1 = 0
77     Led2 = 0
78  End Sub
79
80  'interrupt sequence (triggered by rising
81  'edge of sync-signal on PIN D2 (INT0))
82  Light:
83     Set Portc.4            'set sync out
84     Timer0 = 0
85     Timer1 = 1023
86     Compare1a = Led2
87     Compare1b = Led1
88     Start Timer0
89     Do
90        If Timer0 > 100 Then Exit Do
91     Loop Until Pind.2 = 0
92     Reset Portc.4
93     Compare1a = 0
94     Compare1b = 0
95     Led1 = 1023 - Getadc(0)
96     Led2 = 1023 - Getadc(1)
97     If Led1 > 900 Then     'maximum brightness level
98        Led1 = 900
99     End If
100    If Led2 > 900 Then
101       Led2 = 900
102    End If
103    Turn_off               'check if off-switch is set
104    Timer0 = 0
105 Return
```

A.3 Camera trigger source code

This BASIC source code is required to control the external trigger input of the applied PointGrey Dragonfly Express camera [152]. Again, the BASCOM AVR BASIC compiler [38] has to be used to compile the code.

```
$regfile = M8def.dat
$crystal = 8000000

'pullup reset pin
Set Portc.6

'sync inputs (rising/falling edge)
Config Pind.2 = Input
Config Pind.3 = Input

'pullup sync input pins (then pulled down by optocoupler)
Set Portd.2
Set Portd.3

Config Pind.4 = Output      ' trigger output
Trigger Alias Portd.4
Config Portc = Input        'dip switches
Portc = 63                  'pull up's on

Config Portb = Output       'LED's


Dim N As Byte
Dim Cycles As Byte
Dim A As Byte
Dim B As Byte

Declare Sub Chk_mode()

' configure ext. interrupts
Config Int0 = Rising
Config Int1 = Falling

On Int0 Count0
On Int1 Count1
Enable Int0
Enable Int1
Enable Interrupts

N = 0
Cycles = 1
Portb = 32

Do
Idle
Loop
End

Sub Chk_mode()
Dim Val_ As Byte

    If Pinc.3 = 0 Then
        Config Int0 = Rising
        Config Int1 = Falling
```

```
55      Else
56          Config Int0 = Falling
57          Config Int1 = Rising
58      End If
59
60      If Pinc.2 = 0 Then
61          Portb = 4                                   'activate green status LED
62          Val_ = 3 And Pinc
63          Cycles = Val_ + 1
64          A = 0
65          B = Cycles
66
67      Else
68          If Pinc.0 = 0 Then
69              Portb = 16              'activate yellow status LED
70              Cycles = 3
71              A = 0
72              B = 1
73              Config Int0 = Rising
74              Config Int1 = Falling
75          Else                        'activate red status LED
76               Portb = 32
77               Cycles = 1
78              A = 0
79              B = 0
80          End If
81      End If
82  End Sub
83
84  Count0:
85   If N = A Then
86              Set Trigger
87              Waitus 500
88              Reset Trigger
89   End If
90   Chk_mode
91  Return
92
93  Count1:
94   If N = B Then
95              Set Trigger
96              Waitus 500
97              Reset Trigger
98   End If
99   Incr N
100  N = N Mod Cycles
101  Chk_mode
102 Return
```

A.4 Questionnaire for Evaluating the Derived Δ-Function

The questionnaire applied in the user evaluation to validate our derived function for the adaptive, imperceptible Δ-coded projection presented in chapter 4.1.5 is shown on the following pages.

Allgemeine Informationen über sie

Beruf/Tätigkeit/Studiengang: _____

Alter: _____

Geschlecht: ☐ Weiblich ☐ Männlich

Ihnen werden jetzt eine Reihe von Bildern auf verschiedene Arten präsentiert. Bitte betrachten sie die projizierten Bilder in ruhe und achten sie darauf, ob sie bei der Betrachtung etwas als unangenehm oder störend empfinden (Bitte ankreuzen).
In der linken unteren Ecke wird die Nummer des aktuellen Tests grün eingeblendet.

ES IST NOTWENDIG, DASS SIE IHREN EINDRUCK KURZ BESCHREIBEN, WENN IHNEN IN DER PROJEKTION ETWAS UNANGENEHMES AUFFÄLLT.

Bild 1	**Bemerke nichts störendes (a)**	**Darstellung wirkt leicht unangenehm (b)**	**Darstellung wirkt sehr unangenehm (c)**	**Wenn (b) oder (c) – bitte beschreiben sie den Eindruck**
Test 1:	O	O	O	
Test 2:	O	O	O	
Test 3:	O	O	O	
Test 4:	O	O	O	
Test 5:	O	O	O	
Test 6:	O	O	O	
Test 7:	O	O	O	

Bild 2	**Bemerke nichts störendes (a)**	**Darstellung wirkt leicht unangenehm (b)**	**Darstellung wirkt sehr unangenehm (c)**	**Wenn (b) oder (c) – bitte beschreiben sie den Eindruck**
Test 1:	O	O	O	
Test 2:	O	O	O	
Test 3:	O	O	O	
Test 4:	O	O	O	
Test 5:	O	O	O	
Test 6:	O	O	O	
Test 7:	O	O	O	

bitte wenden…

Bild 3	**Bemerke nichts störendes (a)**	**Darstellung wirkt leicht unangenehm (b)**	**Darstellung wirkt sehr unangenehm (c)**	**Wenn (b) oder (c) – bitte beschreiben sie den Eindruck**
Test 1:	O	O	O	
Test 2:	O	O	O	
Test 3:	O	O	O	
Test 4:	O	O	O	
Test 5:	O	O	O	
Test 6:	O	O	O	
Test 7:	O	O	O	

Ihnen wird jetzt eine Videosequenz auf verschiedene Arten präsentiert. Achten sie bitte darauf, ob sie bei der Betrachtung etwas als unangenehm oder störend empfinden (Bitte ankreuzen):

Video 1	**Bemerke nichts störendes (a)**	**Darstellung wirkt leicht unangenehm (b)**	**Darstellung wirkt sehr unangenehm (c)**	**Wenn (b) oder (c) – bitte beschreiben sie den Eindruck**
Test 1:	O	O	O	
Test 2:	O	O	O	
Test 3:	O	O	O	
Test 4:	O	O	O	
Test 5:	O	O	O	
Test 6:	O	O	O	
Test 7:	O	O	O	

Hier haben sie Platz für Anmerkungen und Kommentare:

_____

_____

_____

_____

_____

_____

Vielen Dank für ihre Teilnahme!

A.5 Curriculum Vitae

The following pages list the curriculum vitae of Anselm Grundhöfer including his scientific services.

Anselm Grundhöfer – Curriculum Vitae

Personal Data

Title:	Diplom Mediensystemwissenschaftler
Name:	Anselm Grundhöfer
Address:	Jahnstr. 26 99423 Weimar Germany
Mobile telephone	+49 177 8234633
Email:	anselmgrundhoefer@gmail.com
Born:	January, 11th, 1980 in Karlsbad-Langensteinbach, Germany
Citizenship:	German

Education

2006 – 2010	Ph.D. candidate and research scientist Bauhaus-Universität Weimar Faculty of media Augmented Reality department
2009	Advanced training "personnel management", certificate
2000 – 2006	Studies of Media Systems Science Bauhaus-Universität Weimar, Diploma with honors
October 2003 – March 2004	Exchange student at Osaka University
1990 – 1999	Frankenwald-Gymnasium Kronach, "Abitur" certificate

Work and Teaching Experience

April '10 – August '10	Lecturer, Johannes-Kepler-University, Linz
2009	Consultant and software developer for Vioso GmbH
October '06 – March '10	Teaching assistant, "Information Displays" and "Computer Vision" Bauhaus- Universität Weimar, Faculty of media, Augmented Reality department
2005 – 2007	Consultant and software developer device+context and Bennert Monumedia GmbH
April '04 – June '06	Research assistant HoloGraphics Project (DFG) Bauhaus-Universität Weimar Faculty of media, Augmented Reality department
August '03 – September '03	Research internship Fraunhofer IGD, Darmstadt
Summer 2005	Research internship Synotec Psychoinformatik GmbH
April '02 – July '03	Research assistant Medienquadrat Project (BMBF), Bauhaus-Universität Weimar
September '99 – July '00	Civil service, Ökologische Bildungsstätte Oberfranken, Mitwitz

Invited Talks

2004	*"Intuitive Graphical User Interfaces for the Electronic Sound Generation based on Genetic Algorithms and Fuzzy Sets"* Harmonie Sensorielle, Paris, France
2006	*"Reverse Radiosity: Compensating Indirect Scattering for Immersive and Semi-Immersive Projection Displays"* University of Otago, Dunedin, New Zealand
2007	*"Dynamic Adaptation of Projected Imperceptible Codes"* International workshop on ambient interfaces, Osaka University, Japan
2008	*"VirtualStudio2Go: Digital Video Composition for Real Environments"* University of British Columbia, Vancouver, Canada
2009	*"VirtualStudio2Go: Digital Video Composition for Real Environments"* Winter Augmented Reality Meeting, TU Graz, Austria
2010	*"Synchronized Projector-Camera Systems for Digital Video Composition"* ETH Zürich, Switzerland

Patents

"Verfahren zur Erzeugung erweiterter Realität in einem Raum"
Bimber, O. Grundhöfer, A., Zollmann, S., and Kolster, D.
German patent DE 10 2007 041 719.7

"Adaptive Coded Aperture Projection"
Bimber, O., Grosse, M., Wetzstein, G., Grundhöfer, A..
German patent application PVA 02/039

"Chromakeyverfahren und Chromakeyvorrichtung zur Aufnahme und Bildbearbeitung von Kamerabildern"
Bimber, O., Grundhöfer, A., Thiele, S., Kurz, D.
German patent application PVA 02/046

Services

2006-2010	Paper reviewer for ISMAR, IEEE VR, IEEE Transactions on Circuits and Systems for Video Technology, Eurographics, ProCams, JVRB, Virtual Reality Journal
2007	Program committee member of the International Workshop on Projector-Camera Systems

Languages

German	mother tongue
English	written and spoken
Japanese/ Spanish	basic knowledge

Awards

2006	1st Place ACM Siggraph Student Research Competition
2007	University Award, Bauhaus-Universität Weimar
	1st Place ACM Student Research Competition

REFERENCES

[1] Science.D.Vision 3DEqualizer. Product web page, 00:32am, 15.08.2010. http://www.sci-d-vis.com/. 84

[2] Aja. Aja xena hd sdi capture card, 4:21pm, 18.05.2010. http://www.aja.com/products/kona/konalhe/. 107

[3] H. Ando, J. Watanabe, T. Amemiya, and T. Maeda. Full-scale saccade-based display: public/private image presentation based on gaze-contingent visual illusion. In *EDT '07: Proceedings of the 2007 workshop on Emerging displays technologies*, page 12, New York, NY, USA, 2007. ACM. 20

[4] M. Ashdown, T. Okabe, I. Sato, and Y. Sato. Robust content-dependent photometric projector compensation. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, page 6, Washington, DC, USA, 2006. IEEE Computer Society. 25, 28

[5] S. Baker, E. Bennett, S.B. Kang, and Szeliski R. Removing rolling shutter wobble. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'10)*, 2010. 96

[6] R.S. Bannister. Chroma keying system. US Patent number 4,319,266, 1982. 30, 35

[7] B.E. Bayer. Color imaging array. US Patent number 3,971,065, 1976. 96

[8] N. Beato, Y. Zhang, M. Colbert, K. Yamazawa, and C.E. Hughes. Interactive chroma keying for mixed reality. *Comput. Animat. Virtual Worlds*, 20(2‐3):405–415, 2009. 30

[9] A. Belmares-Sarabia and S.J. Chayka. Video color detector and chroma key device and method. US Patent number 4,811,084, 1989. 30, 35

[10] M. Ben Ezra. Segmentation with invisible keying signal. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,CVPR*, volume 1, pages 32–37. IEEE Computer Society, 2000. 32

[11] A. Berman, A. Dadourian, and P. Vlahos. Method for removing from an image the background surrounding a selected object. US Patent number 6,134,346, 2000. 33

[12] A. Berman, P. Vlahos, and A. Dadourian. Comprehensive method for removing from an image the background surrounding a selected object. US Patent number 6,134,345, 2000. 33

[13] O. Bimber, A. Emmerling, and T. Klemmer. Embedded entertainment with smart projectors. *Computer*, 38(1):48–55, 2005. 25

[14] O. Bimber, A. Grundhöfer, and S. Wetzstein, G.and Knödel. Consistent illumination within optical see-through augmented environments. In *ISMAR '03: Proceedings of the The 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 198–207, Washington, DC, USA, 2003. IEEE Computer Society. 63, 71

[15] O. Bimber, A. Grundhöfer, S. Zollmann, and D. Kolster. Digital illumination for augmented studios. *Journal of Virtual Reality and Broadcasting*, 3(8), December 2006. urn:nbn:de:0009-6-6302, ISSN 1860-2037. 5, 46, 47, 71

[16] O. Bimber, D. Iwai, G. Wetzstein, and A. Grundhöfer. The Visual Computing of Projector-Camera Systems. In *Proc. Eurographics (State-of-the-Art Report)*, pages 23–46, 2007. 21, 76

[17] O. Bimber, G. Wetzstein, A. Emmerling, and C. Nitschke. Enabling View-Dependent Stereoscopic Projection in Real Environments. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 14–23, 2005. 25

[18] J.R. Bitner, G. Ehrlich, and E.M. Reingold. Efficient generation of the binary reflected gray code and its applications. *Commun. ACM*, 19(9):517–521, 1976. 24, 75

[19] K.R. Boff and J.E. Lincoln. *Engineering Data Compendium: Human Perception and Performance*, volume 1. O'Reilly, AAMRL, Wright-Patterson AFB, OH, 1988. 19

[20] D. Bradley, B. Atcheson, I. Ihrke, and W. Heidrich. Synchronization and rolling shutter compensation for consumer video camera arrays. *Proceedings of the IEEE International Workshop on Projector-Camera Systems (Procams 2009)*, 2009. 96

[21] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly, Cambridge, MA, 2008. 22

[22] M.S. Brennesholtz and E.H. Stupp. *Projection Displays*. Wiley Publishing, 2008. iv, 19, 20, 51, 52

[23] R. Brinkmann. *The art and science of digital compositing*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999. 30

[24] P.J. Burt and E.A. Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, COM-31,4:532–540, 1983. 55

[25] JVC GY-HD251E Camera. Technical specification jvc gy-hd251e camera, 9:18pm, 03.05.2010. `http://pro.jvc.com/prof/attributes/specs.jsp?model_id=MDL101625&feature_id=03`. 96, 107

[26] A. Cernasov. *Digital Video Electronics with 12 Complete Projects*. McGraw-Hill/TAB Electronics, 2004. 18

[27] T.J. Cham, J.M. Rehg, R. Sukthankar, and G. Sukthankar. Shadow elimination and occluder light suppression for multi-projector displays. pages II: 513–520, 2003. 65

[28] D.J. Chaplin. Chroma key method and apparatus. US Patent number 5,249,039, 1993. 30, 35

[29] D.J. Chaplin. Chroma keyer with fringe control offset. US Patent number 5,313,304, 1994. 36

[30] D.J. Chaplin. Chroma keyer with correction for background defects. US Patent number 5,400,081, 1995. 30, 35

[31] Y.-Y. Chaung, D.E. Zongker, J. Hindordd, B. Curless, D.H. Salesin, and R. Szeliski. Environment matting extensions: Towards higher accuracy and real-time capture. In *Proceedings of ACM SIGGRAPH 2000*, Computer Graphics Proceedings, Annual Conference Series, pages 121–130. ACM Press / ACM SIGGRAPH / Addison Wesley Logman, July 2000. ISBN 1-58113-208-5. 79, 80

[32] Yang X. Xiao S. Chen, X. and M. Li. A practical radiometric compensation method for projector-based augmentation. In *ISMAR '08: Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 155–156, Washington, DC, USA, 2008. IEEE Computer Society. 27

[33] B. Choudhury, D. Singla, and S. Chandran. Fast color-space decomposition based environment matting. In *I3D '08: Proceedings of the 2008 symposium on Interactive 3D graphics and games*, pages 1–1, New York, NY, USA, 2008. ACM. 33, 79

[34] Y.-Y. Chuang. *New models and methods for matting and compositing*. PhD thesis, 2004. Chair-Curless, Brian and Chair-Salesin, David H. 15

[35] Y.-Y. Chuang, A. Agarwala, B. Curless, D.H. Salesin, and R. Szeliski. Video matting of complex scenes. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 243–248, New York, NY, USA, 2002. ACM. 33

[36] Y.-Y. Chuang, B. Curless, D.H. Salesin, and R. Szeliski. A bayesian approach to digital matting. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 264–271, Los Alamitos, CA, USA, 2001. IEEE Computer Society. 33, 100

[37] Y.-Y. Chuang, D.E. Zongker, J. Hindorff, B. Curless, D.H. Salesin, and R. Szeliski. Environment matting extensions: towards higher accuracy and real-time capture. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 121–130, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 33

[38] BASCOM compiler. Product webpage, 4:57pm, 19.05.2010. `http://www.mcselec.com`. 117, 119

[39] A.I. Comport, E. Marchand, M. Pressigout, and F. Chaumette. Real-time markerless tracking for augmented reality: The virtual visual servoing framework. *IEEE Transactions on Visualization and Computer Graphics*, 12:615–628, 2006. 41

[40] D. Cotting, M. Naef, M.H. Gross, and H. Fuchs. Imperceptible patterns for reliable acquisition of mixed reality environments. In *Proceedings of The International Workshop on Image Analysis for Multimedia Interactive Services 2005*, 2005. 39

[41] D. Cotting, M. Näf, M.H. Gross, and H. Fuchs. Embedding imperceptible patterns into projected images for simultaneous acquisition and display. In *Proceedings of the Third IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'04)*, pages 100–109, 2004. 39, 41, 49, 51

[42] D. Cotting, R. Ziegler, M.H. Gross, and H. Fuchs. Adaptive instant displays: Continuously calibrated projections using per-pixel light control. In *Proceedings Eurographics 2005*, pages 705–714, 2005. Eurographics 2005, Dublin, Ireland, August 29 - September 2, 2005. 39

[43] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 369–378, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co. 108

[44] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, and T. Hawkins. A lighting reproduction approach to live-action compositing. *ACM Trans. Graph.*, 21(3):547–556, 2002. 4, 3, 30, 38, 112

[45] J.A. Delwiche. Chroma keyer with secondary hue selector. US Patent number 5,251,016, 1993. 30, 35

[46] DepthQ. Depthq webpage, 6:33pm, 17.05.2010. `http://www.depthq.com/index.html`. 61, 106

[47] P. Didyk, E. Eisemann, T. Ritschel, K. Myszkowski, and H.-P. Seidel. Apparent display resolution enhancement for moving images. *ACM Transactions on Graphics (Proceedings SIGGRAPH 2010, Los Angeles)*, 29(3), 2010. 20

[48] J. Dupont and F. Deschenes. Toward a realistic interpretation of blue-spill for blue-screen matting. In *CRV '06: Proceedings of the The 3rd Canadian Conference on Computer and Robot Vision*, page 33, Washington, DC, USA, 2006. IEEE Computer Society. 36, 89

[49] dvGarage. dvmatte pro website, 6:22pm, 17.07.2010. `http://www.dvgarage.com/dvmatte-pro`. 78

[50] Osram LED Ostar LE W E3B. Technical specification osram ostar led le w e3b, 2:21pm, 03.05.2010. `http://catalog.osram-os.com/catalogue/catalogue.do?favOid=00000001000161360007d0023&act=showBookmark`. 73

[51] M. Fiala. Artag, a fiducial marker system using digital techniques. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 590–596, Washington, DC, USA, 2005. IEEE Computer Society. 64, 66, 68, 70

[52] M. Fiala. Automatic projector calibration using self-identifying patterns. In *Proceedings of the IEEE International Workshop on Projector-Camera Systems (Procams 2005)*, San Diego, USA, 2005. 42, 76

[53] D. Fofi, T. Sliwa, and Y. Voisin. A comparative survey on invisible structured light. volume 5303, pages 90–98. SPIE, 2004. 24, 41

[54] P.-E. Forssen and E. Ringaby. Rectifying rolling shutter video from hand-held devices. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'10)*, 2010. 96

[55] J.-M. Frahm, K. Koeser, D. Grest, and R. Koch. Markerless augmented reality with light source estimation for direct illumination. In *Conference on Visual Media Production CVMP, London, December 2005*, 2005. 41

[56] M. Fuchs, V. Blanz, and H.-P. Seidel. Bayesian relighting. In Oliver Deussen, Alexander Keller, Kavita Bala, Philip Dutré, Dieter W. Fellner, and Stephen N. Spencer, editors, *Rendering Techniques 2005: Eurographics Symposium on Rendering*, Rendering Techniques, pages 157–164, Konstanz, Germany, July 2005. Eurographics. 112

[57] K. Fujii, M.D. Grossberg, and S.K. Nayar. A projector-camera system with real-time photometric adaptation for dynamic environments. pages I: 814–821, 2005. 25

[58] T. Fukaya, H. Fujikake, Y. Yamanouchi, H. Mitsumine, N. Yagi, S. Inoue, and H. Kikuchi. An effective interaction tool for performance in the virtual studio - invisible light projection system. In *Proceedings International Broadcasting Conference (IBC03)*, pages 389–396, Japan, 2003. 38, 83

[59] T. Fukaya, T. Iwai, and Y. Yamanouchi. Morphovision. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Emerging technologies*, page 22, New York, NY, USA, 2006. ACM. 3, 2

[60] S. Gaurav. *Digital Color Imaging Handbook*. CRC Press, Inc., Boca Raton, FL, USA, 2002. 100

[61] Black Magic Mini Sync Black Burst Signal Generator. Technical specification, 11:42pm, 15.08.2010. `http://www.blackmagic-design.com/products/miniconverters/techspecs/`. 71

[62] J. Gühring. Dense 3-d surface acquisition by structured light using off-the-shelf components. In *Proc. Videometrics and Optical Methods for 3D Shape Measurement*, pages 220–231, 2001. 75

[63] O. Grau, T. Pullen, and G.A. Thomas. A combined studio production system for 3-d capturing of live action and immersive actor feedback. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(3):370–380, 2004. 4, 3, 38

[64] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A.V. Moere, and O. Staadt. blue-c: a spatially immersive display and 3d video portal for telepresence. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 819–827, New York, NY, USA, 2003. ACM. 38

[65] M.D. Grossberg, H. Peri, S.K. Nayar, and P.N. Belhumeur. Making one object look like another: Controlling appearance using a projector-camera system. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04)*, 1, 2004. 27, 76

[66] A. Grundhöfer and O. Bimber. Dynamic bluescreens. In *SIGGRAPH '08: ACM SIGGRAPH 2008 talks*, pages 1–1, New York, NY, USA, 2008. ACM. 38, 73

[67] A. Grundhöfer and O. Bimber. Real-time adaptive radiometric compensation. *IEEE Transactions on Visualization and Computer Graphics*, 14:97–108, 2008. 25, 66

[68] A. Grundhöfer and O. Bimber. Virtualstudio2go: digital video composition for real environments. In *SIGGRAPH Asia '08: ACM SIGGRAPH Asia 2008 papers*, pages 1–8, New York, NY, USA, 2008. ACM. 38, 73, 100, 103

[69] A. Grundhöfer, D. Kurz, S. Thiele, and O. Bimber. Color invariant chroma keying and color spill neutralization for dynamic scenes and cameras. In *The Visual Computer*, volume 26, Springer Berlin / Heidelberg, 2010. Springer. 87

[70] A. Grundhöfer, M. Seeger, F. Hantsch, and O. Bimber. Dynamic adaptation of projected imperceptible codes. In *ISMAR '07: Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–10, Washington, DC, USA, 2007. IEEE Computer Society. 34

[71] C.L. Guo, Q. Ma, and L.M. Zhang. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. pages 1–8, 2008. 60

[72] C.L. Guo and L.M. Zhang. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. 19(1):185–198, January 2010. 60

[73] R. Gvili, A. Kaplan, E. Ofek, and G. Yahav. Depth keying. In Andrew J. Woods, Mark T. Bolas, John O. Merritt, and Stephen A. Benton, editors, *SPIE Stereoscopic Displays and Virtual Reality Systems X*, volume 5006, pages 564–574. SPIE, 2003. 32, 35, 36, 63

[74] Ultimatte 11 HD/SD. Product webpage, 12:14pm, 27.07.2010. `http://www.ultimatte.com/UltimatteMain/11_Main.html`. 97

[75] W. A. Hershberger. Saccadic eye movements and the perception of visual direction. In *Perception and Psychophysics*, 1987. 20, 51

[76] W. A. Hershberger. The phantom array: a perisaccadic illusion of visual direction. In *The Psychological Record*. Psychological Record, 1998. 20, 51

[77] N. Hideaki, S. Maki, and I. Masahiko. Smart light-ultra high speed projector for spatial multiplexing optical transmission. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, page 95, Washington, DC, USA, 2005. IEEE Computer Society. 113

[78] P. Hillman, J. Hannah, and D. Renshaw. Alpha channel estimation in high resolution images and image sequences. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 1, page 1063, Los Alamitos, CA, USA, 2001. IEEE Computer Society. 33

[79] Xbox360 You're in the Movies. compositing game web page, 7:16pm, 16.08.2010. `http://www.xbox.com/de-DE/games/y/yitm/`. 113

[80] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998. 60

[81] M.E. Jaen. Temporal lighting modulation and visual information coding mechanisms. *Perception - ECVP*. 20

[82] A. Jones, A. Gardner, M. Bolas, I. McDowall, and P. Debevec. A lighting reproduction approach to live-action compositing. *Visual Media Production, 2006. CVMP 2006. 3rd European Conference on*, pages 127–133, 2006. 38, 112

[83] P.K. Kaiser. The joy of visual perception - blochs law, 7:42pm, 01.07.2010. `http://www.yorku.ca/eye/blochlaw.htm`. 18

[84] H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *IWAR '99: Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality*, page 85, Washington, DC, USA, 1999. IEEE Computer Society. 70

[85] D.H. Kelly. Visual response to time-dependent stimuli. In *I. Amplitude sensitivity measurements. J. Opt. Soc.*, 1961. 49

[86] F. Kelly and A. Kokaram. Fast image interpolation for motion estimation using. In *in Real Time Imaging VIII, SPIE*, pages 184–194, 2004. 112

[87] E.A. Khan, E. Reinhard, R.W. Fleming, and H.H. Bülthoff. Image-based material editing. *ACM Trans. Graph.*, 25(3):654–663, 2006. 82

[88] Khronos. The opencl specification 1.0, 2:13pm, 12.07.2010. `www.khronos.org/registry/cl/specs/opencl-1.0.29.pdf`. 112

[89] J. Kim, M. Hwangbo, and T. Kanade. Realtime affine-photometric klt feature tracker on gpu in cuda framework. In *Workshop on Embedded Computer Vision (ECV), 2009 (held in conjunction with ICCV)*, October 2009. 112

[90] R. Koch, I. Schiller, B. Bartczak, F. Kellner, and K. Koeser. Mixin3d: 3d mixed reality with tof-camera. In *Dynamic 3D Imaging DAGM 2009 Workshop, Dyn3D, LNCS 5742*, pages 126–141, Jena, Germany, September 2009. 32

[91] H Kolb, E. Fernandez, and R Nelson. Webvision the organization of the vertebrate retina., 7:39pm, 01.07.2010. `http://webvision.med.utah.edu/temporal.html`. 18

[92] Krylon. Krylon industrial eco-guard digital colors chroma-key paint, 7:13pm, 12.08.2010. `http://www.kpg-industrial.com/products/ecoguard_digital_colors/`. 30

[93] D. Kurz, F. Hantsch, M. Grosse, A. Schiewe, and O. Bimber. Laser pointer tracking in projector-augmented architectural environments. In *ISMAR*, pages 19–26, 2007. 63

[94] M.-H. Lee, B.K. Seo, and J.-I. Park. Change-of-lighting compensation for intelligent image projection. *The 15th Japan-Korea Joint Workshop on Frontiers of Computer Vision*, 15, 2009. 25

[95] S.-Y. Lee, I.-J. Kim, S.C. Ahn, and H.-G. Kim. Active segmentation for immersive live avatar. *Electronics Letters*, 40:1257–1258, 2004. 30

[96] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 775–781, Washington, DC, USA, 2005. IEEE Computer Society. 70

[97] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1:61–68, 2006. 33

[98] H. Levkowitz. *Color Theory and Modeling for Computer Graphics, Visualization, and Multimedia Applications*. Kluwer Academic Publishers, Norwell, MA, USA, 1997. 100

[99] C.-K. Lin. Pixel grouping for color filter array demosaicing. http://web.cecs.pdx.edu/~cklin/demosaic/ (last viewed: 25/05/2007), 2006. 68

[100] J. Lubin. A visual discrimination model for imaging system design and evaluation. *Vision Models for target detection and recognition*, pages 245–283, 1995. 56

[101] R. Lukac and K.N. Plataniotis. Color filter arrays: design and performance analysis. *IEEE Transactions on Consumer Electronics*, 51:1260–1267, 2005. 96

[102] R. Lukac and K.N. Plataniotis. *Color Image Processing: Methods and Applications*. CRC Press; 1 edition, 2006. 96

[103] K.H. Madsen. *Production methods: behind the scenes of virtual inhabited 3D worlds*. Springer-Verlag New York, Inc., New York, NY, USA, 2003. 37

[104] S Martinez-Conde, S.L. Macknik, L.M. Martinez, and Alonso J.M. Visual perception (part i). fundamentals of vision: low and mid-level processes in perception. *Progress in Brain Research Series*. 20

[105] I.E. McDowall, M.T. Bolas, P. Hoberman, and S.S. Fisher. Snared illumination. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Emerging technologies*, page 24, New York, NY, USA, 2004. ACM. 20

[106] M. McGuire and W. Matusik. Real-time triangulation matting using passive polarization. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches*, page 88, New York, NY, USA, 2006. ACM. 32

[107] M. McGuire, W. Matusik, H. Pfister, J.F. Hughes, and F. Durand. Defocus video matting. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 567–576, New York, NY, USA, 2005. ACM. 31

[108] Y. Mishima. Soft edge chroma-key generation based upon hexoctahedral color space. US Patent number 5,355,174, 1994. 30, 35

[109] H. Mitsumine, T. Fukaya, S. Komiyama, and Y. Yamanouchi. Immersive virtual studio. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Sketches*, page 121, New York, NY, USA, 2005. ACM Press. 4, 3

[110] R.A. Moses and W.M. Hart. The temporal responsiveness of vision. In *Adler's Physiology of the Eye*, 1987. v, 49, 55, 57, 100

[111] M. Moshkovitz. *The virtual studio*. Focal Press, USA, 2000. 37

[112] J. Nakamura. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, Inc., Boca Raton, FL, USA, 2005. 96

[113] S.G. Narasimhan, S.J. Koppal, and Shuntaro Y. Temporal dithering of illumination for fast active vision. In *Proc. European Conference on Computer Vision*, pages 830–844, October 2008. 39

[114] S. K. Nayar, H. Peri, M. D. Grossberg, and P. N. Belhumeur. A Projection System with Radiometric Compensation for Screen Imperfections. In *Proc. of IEEE International Workshop on Projector-Camera Systems (ProCams)*, 2003. 76

[115] P.D. Netzley. *Encyclopedia Of Movie Special Effects*. Greenwood, 1999. 2, 1

[116] A. Noritake, Wantanabe J., H. Ando, M. Terao, and A. Yagi. Spatial independency in perceived lengths of saccade-induced images. In *Psychologia*, 2005. 20

[117] Nvidia. Quadro sdi capture product specification, 2010. `http://www.nvidia.com/object/product_quadro_sdi_capture_us.html`. 97

[118] Nvidia. The cuda programming guide 3.1, 2:16pm, 12.07.2010. `http://developer.download.nvidia.com/compute/cuda/3_1/toolkit/docs/NVIDIA_CUDA_C_ProgrammingGuide_3.1.pdf`. 112

[119] Nvidia. Quadro g-sync 2 user guide, 4:38pm, 18.05.2010. http://www.nvidia.com/docs/IO/40049/Quadro_GSync_5800_4800_install_guide.pdf. 73, 106, 107

[120] OpenCV. Open computer vision library, 5:26pm, 17.05.2010. http://sourceforge.net/projects/opencvlibrary/. 66

[121] H. Park, M.H. Lee, B.K.S.Y. Jin, and J.I. Park. Content adaptive embedding of complementary patterns for nonintrusive direct-projected augmented reality. In *HCI International 2007*, volume 14, 2007. 40, 41

[122] H. Park, M.H. Lee, B.K. Seo, J.I. Park, M.S. Jeong, T.S. Park, Y. Lee, and S.R. Kim. Simultaneous geometric and radiometric adaptation to dynamic surfaces with a mobile projector-camera system. 18(1):110–115, January 2008. 40

[123] H. Park, M.H. Lee, B.K. Seo, H.C. Shin, and J.I. Park. Radiometrically-compensated projection onto non-lambertian surface using multiple overlapping projectors. pages 534–544, 2006. 25

[124] H. Park and J.I. Park. Invisible marker tracking for ar. In *ISMAR '04: Proceedings of the Third IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'04)*, pages 272–273, Washington, DC, USA, 2004. IEEE Computer Society. 42

[125] H. Park, B.K. Seo, and J.-I. Park. A nonintrusive method for generating all-focused projection. *Proceedings of IEEE International Conference on Image Processing (ICIP'08)*, 15:529–532, 2008. 40

[126] S.N. Pattanaik, J.A. Ferwerda, M.D. Fairchild, and D.P. Greenberg. A multiscale model of adaptation and spatial vision for realistic image display. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 287–298, New York, NY, USA, 1998. ACM Press. 56

[127] P. Peers and P. Dutré. Wavelet environment matting. In *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, pages 157–166, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. 33, 79

[128] PixelQI. Product web page, 11:18am, 18.05.2010. http://www.pixelqi.com/products. 95

[129] T. Porter and T. Duff. Compositing digital images. In *SIGGRAPH '84: ACM SIGGRAPH 1984 Papers*, pages 253–259, New York, NY, USA, 1984. ACM. 15

[130] D.L. Post, A.L. Nagy, P. Monnier, and C.S. Calhoun. Predicting color breakup on field-sequential displays: Part 2. *SID International Symposium Digest of Technical Papers*, pages 1037–1040, 1998. 51

[131] Black Magic Multibridge Pro. Technical specification, 12:13pm, 27.07.2010. `http://www.blackmagic-design.com/products/multibridge/techspecs/`. 97

[132] R.J. Qian and M.I. Sezan. Video background replacement without a blue screen. pages IV:143–146, 1999. 31

[133] M. Ramasubramanian, S.N. Pattanaik, and D.P. Greenberg. A perceptually based physical error metric for realistic image synthesis. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 73–82, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. v, 55, 57, 58

[134] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs. The office of the future: A unified approach to image-based modeling and spatially immersive displays. *Computer Graphics*, 32(Annual Conference Series):179–188, 1998. 39, 40, 47, 51

[135] R. Raskar, G. Welch, K.-L. Low, and D. Bandyopadhyay. Shader lamps: Animating real objects with image-based illumination. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques, London, UK, June 25-27, 2001*, pages 89–102, 2001. ISBN 3-211-83709-4. 63, 71

[136] E. Reinhard and E.A. Khan. Depth-of-field-based alpha-matte extraction. In *APGV '05: Proceedings of the 2nd symposium on Applied perception in graphics and visualization*, pages 95–102, New York, NY, USA, 2005. ACM Press. 63, 70

[137] E. Reinhard, E.A. Khan, A.O. Akyz, and G.M. Johnson. *Color Imaging: Fundamentals and Applications*. A. K. Peters, Ltd., Natick, MA, USA, 2008. 98

[138] R. Rickitt. *Special Effects: The History and Technique*. Billboard Books, 2007. 1

[139] Rosco. Rosco ultimatte chroma-key paint, 7:02pm, 12.08.2010. `http://www.rosco.com/us/scenic/chroma_key.cfm`. 30

[140] C. Rother, V. Kolmogorov, and A. Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. 23(3):309–314, 2004. 33

[141] Mark A. Ruzon and Carlo Tomasi. Alpha estimation in natural images. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1:1018, 2000. 33

[142] J. Salvi, J. Pages, and J. Battle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827 – 849, 2004. 22, 24

[143] M. Sawicki. *Filming the Fantastic: A Guide to Visual Effects Cinematography*. Focal Press, 2007. 2

[144] J.A. Selan. *Merging live video with synthetic imagery*. PhD thesis, 2003. 2, 1

[145] P. Sen, B. Chen, G. Garg, S.R. Marschner, Mark Horowitz, Marc Levoy, and Hendrik P. A. Lensch. Dual Photography. In *Proc. of ACM SIGGRAPH*, pages 745–755, 2005. 28

[146] B.K. Seo, M.-H. Lee, H. Park, S.-K. Young, and J.-I. Park. Direct-projected ar based interactive user interface for medical surgery. *Proceedings of International Conference for Artificial Reality and Telexistence (ICAT'07)*, pages 105–112, 2007. 40

[147] G. Sharma. *Digital Color Imaging Handbook*. CRC Press, Inc., Boca Raton, FL, USA, 2002. 17

[148] A. Shirai, M. Takahashi, K. Kobayashi, H. Mitsumine, and S. Richir. Lumina studio: Supportive information display for virtual studioenvironments. *IEEE VR 2005 Workshop on Emerging Display Technologies*, pages 17–20, 2005. 38

[149] D. Shreiner. *OpenGL programming guide*. Addison-Wesley, Upper Saddle River, NJ, 2006. 66

[150] K. Simon. *Farbe im Digitalen Publizieren: Konzepte der digitalen Farbwiedergabe für Office, Design und Software*. Springer, Berlin, Germany, 2007. 18

[151] A.R. Smith and J.F. Blinn. Blue screen matting. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 259–268, New York, NY, USA, 1996. ACM. viii, 30, 35, 36, 37, 89, 91

[152] PointGrey Dragonfly Express Technical Specification. Technical specification dragonfly express, 9:21pm, 03.05.2010. `http://www.ptgrey.com/support/downloads/documents/Dragonfly20Express20Getting20Started20Manual.pdf`. 106, 119

[153] Datacolor Spyder3Elite. Product webpage, 1:47pm, 03.05.2010. http://www.datacolor.eu/de/produkte/monitor-kalibrierung/spyder3elite/index.html. 108

[154] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum. Poisson matting. In *SIGGRAPH '04: Proceedings of the 31th annual conference on Computer graphics and interactive techniques*, volume 23, pages 315–321, New York, NY, USA, 2004. ACM. 33

[155] J. Sun, Y. Li, S.B. Kang, Z.-B. Xu, X. Tang, and H.-Y. Shum. Flash cut: Foreground extraction with flash and no-flash image pairs. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2007. 31

[156] ORAD Pattern Recognition Optical Camera Tracking System. Product webpage, 1:52pm, 12.07.2010. http://www.orad.co.il/en/page.asp?id=82. 42

[157] K. Tateno, I. Kitahara, and Y. Ohta. A nested marker for augmented reality. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches*, page 152, New York, NY, USA, 2006. ACM Press. 42

[158] Projection Design F32 technical specifications. Product webpage, 9:57pm, 19.07.2010. http://www.projectiondesign.com/Default.asp?CatID=1721. 71

[159] G.A. Thomas, J. Jin, and C. A. Urquhart. A versatile camera position measurement system for virtual reality tv production. In *International Broadcasting Convention*, pages 284–289, 1997. ISBN 0-85296-694-6. 5, 4, 41, 42

[160] G.A. Thomas and R. Russel. Flash-based keying. *European Patent Application EP1499117*, 2005. 31, 35, 62

[161] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998. 80

[162] B. Tomlinson. Using human acting skill to measure empathic ability in heterogeneous characters. *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), Workshop on Empathic Agents. New York, NY.*, 2004. 2

[163] Truematte. retro-reflective chroma-key system, 2:20pm, 08.08.2010. http://www.bbc.co.uk/rd/projects/virtual/truematte. 4, 3

[164] M.B. Vieira, L. Velho, A. Sa, and P.C. Carvalho. A camera-projector system for real-time 3d video. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition (CVPR'05) - Workshops, page 96, Washington, DC, USA, 2005. IEEE Computer Society. 39, 71

[165] Visionresearch. Phantom v710 product page, 3:51pm, 12.07.2010. `http://www.visionresearch.com/index.cfm?sector=htm/files&page=Phantom_v710`. 113

[166] P. Vlahos. Electronic composite photography with color control. US Patent number 4,007,487, 1977. 30, 35, 36, 87, 89

[167] P. Vlahos. Backing color and luminance nonuniformity compensation for linear image compositing. US Patent number 5,032,901, 1991. 36

[168] O. Wada, J. Nakamura, K. Ishikawa, and T. Hatada. Analysis of color breakup in field-sequential color projection system for large area displays. *Proceedings of the Sixth Intl. Display Workshops (IDW99), Japan*, pages 993–996, 1999. 51

[169] D. Wagner and D. Schmalstieg. Artoolkitplus for pose tracking on mobile devices. In *Proceedings of 12th Computer Vision Winter Workshop (CVWW'07)*, 2007. 70

[170] D. Wang, I. Sato, T. Okabe, and Y. Sato. Radiometric compensation in a projector-camera system based properties of human vision system. pages III: 100–100, 2005. 25

[171] M. Waschbüsch, S. Würmlin, D. Cotting, F. Sadlo, and M.H. Gross. Scalable 3d video of dynamic scenes. *The Visual Computer*, 21(8-10):629–638, 2005. 39, 71

[172] J. Watanabe, H. Ando, T. Maeda, and S. Tachi. Gaze-contingent visual presentation based on remote saccade detection. *Presence: Teleoper. Virtual Environ.*, 16(2):224–234, 2007. 20

[173] N. Welsch and C.C. Liebmann. *Farben: Natur, Technik, Kunst*. Spektrum Akademischer Verlag, 2007. 16

[174] A. Wenger, A. Gardner, C. Tchou, J. Unger, T. Hawkins, and P. Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *ACM Trans. Graph.*, 24(3):756–764, 2005. 43, 95, 100, 112

[175] G. Wetzstein and O. Bimber. Radiometric Compensation through Inverse Light Transport. *Proc. of Pacific Graphics*, 2007. 25, 28

[176] Y. Wexler, A.W. Fitzgibbon, and A. Zisserman. Image-based environment matting. In *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*, pages 279–290, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association. 33

[177] T. Whitesides, M. Walls, R. Paolini, S. Sohn, H. Gates, M. McCreary, and J. Jacobson. 10.2: Towards video-rate microencapsulated dual-particle electrophoretic displays. *SID Symposium Digest of Technical Papers*, 35(1):133–135, 2004. 95

[178] Wikipedia. Schematic diagram of the human eye, 5:04pm, 01.07.2010. `http://en.wikipedia.org/wiki/Eye`. iv, 17

[179] L. Williams. Casting curved shadows on curved surfaces. *SIGGRAPH Comput. Graph.*, 12(3):270–274, 1978. 65

[180] A. Wojdala, M. Gruszsewski, and R. Olech. Real-time shadow casting in virtual studio. *Machine Graphics and Vision (MGV)*, 9(1/2):315–329, 2002. 38

[181] S. Wright. *Compositing Visual Effects: Essentials for the Aspiring Artist*. Focal Press, 2007. 35

[182] Y.S. Xirouhakis, A.I. Drosopoulos, and A.N. Delopoulos. Efficient optical camera tracking in virtual sets. *IEEE Transactions on Image Processing*, 10(4):609–622, 2001. ISSN 1057-7149. 42

[183] K. Yasuda, T. Naemura, and H. Harashima. Thermo-key: Human region segmentation from video. *IEEE Computer Graphics and Applications*, 24:26–30, 2004. 31

[184] H. Yee, S. Pattanaik, and D.P. Greenberg. Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. In *ACM Transactions on Graphics*, pages 39–65. ACM Press, 2001. 60

[185] A. Yohso and K. Ukai. How color break-up occurs in the human visual system: mechanism of color break-up phenomenon. *SID Intl. Symp. Digest of Technical Papers, XXXVII, Paper 25.2*, pages 1225–1228, 2006. 20

[186] T. Yoshida, C. Horii, and K. Sato. A Virtual Color Reconstruction System for Real Heritage with Light Projection. In *Proc. of International Conference on Virtual Systems and Multimedia (VSMM)*, pages 161–168, 2003. 28, 76

[187] C. Zach, D. Gallup, and J.M. Frahm. Fast gain-adaptive klt tracking on the gpu. In *CVPR Workshop on Visual Computer Vision on GPU's (CVGPU)*, pages 1–7, 2008. 112

[188] M.-T. Zhao, S.-J. Xiao, X.-B. Yang, and L.-Z. Ma. Variable refractive index in environment matte. *Journal of Zhejiang University - Science A*, 2006. 33

[189] J. Zhu and Y.-H. Yang. Frequency-based environment matting. In *PG '04: Proceedings of the Computer Graphics and Applications, 12th Pacific Conference*, pages 402–410, Washington, DC, USA, 2004. IEEE Computer Society. 33, 79

[190] S. Zollmann and O. Bimber. Imperceptible calibration for radiometric compensation. In *In Proceedings Eurographics 2007, Short Paper*, 2007. 40, 41, 51

[191] D.E. Zongker, D.M. Werner, B. Curless, and D.H. Salesin. Environment matting and compositing. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 205–214, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. 33, 79, 100