Free View Point Video - der nächste Schritt in der Unterhaltungselektronik

P. Eisert, P. Kauff, A. Smolic, R. Schäfer

Fraunhofer Heinrich-Hertz-Institut

eisert/kauff/smolic/schaefer@hhi.de

1. Einleitung

Free Viewpoint Video (FVV) bezeichnet eine neue Form digitaler Medien, an der während der letzten Jahre weltweit intensiv geforscht und entwickelt wird. Der Benutzer bekommt dabei die Möglichkeit, sich innerhalb audio-visueller Szenen frei zu bewegen und jeden beliebigen Standpunkt und Blickwinkel einzunehmen. Diese Funktionalität ist bekannt aus der Computer Graphik (virtuelle Welten, Spiele), wobei die Szenen meist jedoch rein virtuell sind oder nur statische oder 2D Ansichten realer Objekte enthalten. Im Gegensatz dazu geht es bei FVV um die interaktive Darstellung realer und dynamischer Szenen, wie sie von Kameras aufgezeichnet werden.

FVV eignet sich am besten für Szenen, bei denen sich das Geschehen auf einer begrenzten Fläche (bzw. Volumen) abspielt. Speziell Sportveranstaltungen mit begrenztem Spielfeld und Stadion sind hierfür ideal geeignet. Das betreffende Gebiet kann mit einer Anzahl *N* Kameras umgeben und zeitsynchron aufgezeichnet werden. Aus den Kamerasignalen und zugehöriger Kalibrierungsinformation (Position, Ausrichtung, interne Kameraparameter) kann eine dynamische 3D Repräsentation der Szene rekonstruiert werden, die dem Benutzer mit einem entsprechenden Player die Funktionalität der freien Navigation bietet.

Im Allgemeinen steigt die erzielbare Qualität mit der Anzahl der Kameras. Auf der anderen Seite steigen auch die Kosten und der Aufwand zur Realisierung eines solchen Systems, d.h. für eine gegebene Aufgabe muss jeweils ein geeigneter Kompromiss gefunden werden.

Die komplette Verarbeitungskette zu FVV besteht prinzipiell aus den Elementen Multiview-Aufnahme (inkl. Kalibrierung), ggf. 3D Rekonstruktion, Repräsentation, Codierung, Übertragung, interaktives Rendering und Wiedergabe. Im Rahmen dieser Veröffentlichung wir ein Teil dieser Technologien erläutert. In Kapitel 2 wird auf die Aufnahme mit Multi-Kamerasystemen eingegangen und in Kapitel 3 werden die verschiedenen Repräsentationsformen erläutert. Kapitel 4 und 5 beschreiben verschiedene am Fraunhofer HHI entwickelte Algorithmen und Systeme. In Kapitel 6 wird ein evolutionäres Konzept für FVV vorgestellt, das am Fraunhofer HHI verfolgt wird. Schließlich folgt in Kapitel 7 ein Ausblick.

2. Multikamera-Aufnahme

Das klassische FVV-Aufnahmesystem wurde von Prof. Takeo Kanade an der Carnegie Mellon Universität in Pittsburgh entwickelt [1] und als kommerzielles System (Eye-Vision) für verschiedene Sportveranstaltungen verwendet.

Das System besteht aus einem Ring von synchronsierten Kameras (Bild 1), die rund um ein Spielfeld angeordnet aufgestellt werden. Eine Master-Kamera steuert das Gesamtsystem und alle anderen Kameras folgen der Masterkamera, wobei Brennweite und Zoom automatisch nachgeführt werden. Allerdings sind die Erfahrungen mit diesem System im praktischen Einsatz sehr unbefriedigend. Zum einen sind der Kalibrierungsaufwand und damit die Betriebskosten immens, zum anderen ist das System sehr anfällig. So wurde z.B. berichtet, dass das System durch vom Wind erzeugte Bewegungen der Stadionaufhängung, an der die Kameras befestigt waren, total dekalibriert wurde und deshalb nur sehr schlechte Ergebnisse lieferte. Aus diesen Gründen hat sich das System trotz der teilweise sehr eindrucksvollen Bilder bisher nicht durchsetzen können.

Bild 1: Reihe mit synchronisierten Kameras des Eye-Vision-Systems

Moderne Konzepte gehen daher von starren Kameras mit fester Brennweite aus, wobei die gesamte nachfolgende Verarbeitung voll elektronisch erfolgt und somit die Nachteile des EyeVision-Systems vermieden werden. Bild 7 zeigt Beispiele, die am Fraunhofer HHI aufgebaut wurden.

Die notwendigen Signalverarbeitungsformen bestehen prinzipiell aus folgenden Schritten:

- Geometrische und photometrische Kalibrierung der einzelnen Kameras,
- Schätzung der geometrischen Zusammenhänge zwischen den einzelnen Ansichten,

- Suche nach Korrespondenzen in den Einzelansichten mittels Korrelation von korrespondierenden Punkten in zwei oder mehreren Ansichten,
- Lokalisierung der korrespondierenden 3D-Punkte.

3. Szenenrepräsentationen

Verschiedene Technologien stehen für Aufnahme, Verarbeitung, Repräsentation und Rendering von FVV zur Verfügung [2]. Alle diese Ansätze verwenden jedoch als Input Multiview-Aufnahmen, d.h. mehrere synchronisierte Ansichten derselben realen Szene von verschienen Standpunkten und Blickrichtungen aus. Die multiplen Kamerasignale werden verarbeiten und in eine spezifische 3D Szenenrepräsentation transformiert, die es erlaubt, virtuelle Ansichten zwischen den real existierenden Kamerapositionen zu erzeugen. Damit wird es dem Benutzer ermöglicht, frei in der Szene zu navigieren (in praktischen Grenzen), d.h. einen beliebigen Standpunkt und Blickwinkel einzunehmen.

Die Wahl eines Formats zur 3D Szenenrepräsentation ist von zentraler Bedeutung für das Design eines FVV Systems. Auf der einen Seite legt es die Anforderungen für die Aufnahme und Multiview-Verarbeitung fest. So erfordert der Einsatz einer rein bildbasierten Repräsentation (s. u.) die Verwendung sehr vieler Kameras. Ein modellbasierter Ansatz (s. u.) erfordert den Einsatz komplexer und inhärent fehlerbehafteter Algorithmen der Bildverarbeitung (z.B. Segmentierung, 3D Rekonstruktion, Tiefenschätzung). Auf der anderen Seite legt die Repräsentation weitgehend die Render-Algorithmen (und damit Navigationsbereich, Qualität, etc.), Interaktivität und falls erforderlich auch Kompression und Übertragung fest.

In der Computergrafik werden Methoden zur 3D Szenenrepräsentation oft als Kontinuum zwischen 2 Extremen dargestellt (Bild 2) [5]. Auf der einen Seite stehen die klassischen 3D Drahtgittermodelle (geometry-based). 3D Objekte werden als 3D Oberfläche mit assoziierter Textur, Farbe und weiteren Attributen (Reflektionseigenschaften, etc.) dargestellt. Dieser Ansatz ist sehr weit verbreitet und die Qualität kann für rein computergenerierte Modelle exzellent sein. Leistungsfähige und billige Hardware sowie Software APIs stehen zur Verfügung. Ein Nachteil dieses Ansatzes ist der hohe Aufwand zur Erzeugung realistischer virtueller Umgebungen, insbesondere für dynamische Szenen. Die automatische Rekonstruktion realer Szenen, wie es für FVV notwendig ist, erfordert komplexe und fehlerbehaftete Algorithmen der Bildverarbeitung.

Das andere Extrem bilden Methoden, die keinerlei Geometrieinformation verwenden (image-based). Hierbei werden virtuelle Zwischenansichten rein durch Interpolation aus den vorhandenen realen Bildern erzeugt. Der Vorteil der rein bildbasierten Methoden liegt darin, dass keine 3D Rekonstruktion notwendig ist. Dafür werden jedoch für eine hohe Qualität sehr viele Bilder benötigt, d.h. sehr viele Kameras, bzw. der Navigationsbereich ist sehr eingeschränkt. Beispiele für rein bildbasierte Ansätze sind Ray-Space [3] und Ligh-Fields [6], sowie Concentric Mosaics [10].

Zwischen den beiden Extremen liegt ein Kontinuum von Technologien, die in verschiedener Weise Geometrie und Bilder vereinen (z.B. Lumigraph [11], [12],

Layered Depth Images [13], View-dependent Texture Mapping [9], Surface Light-Fields [14]).

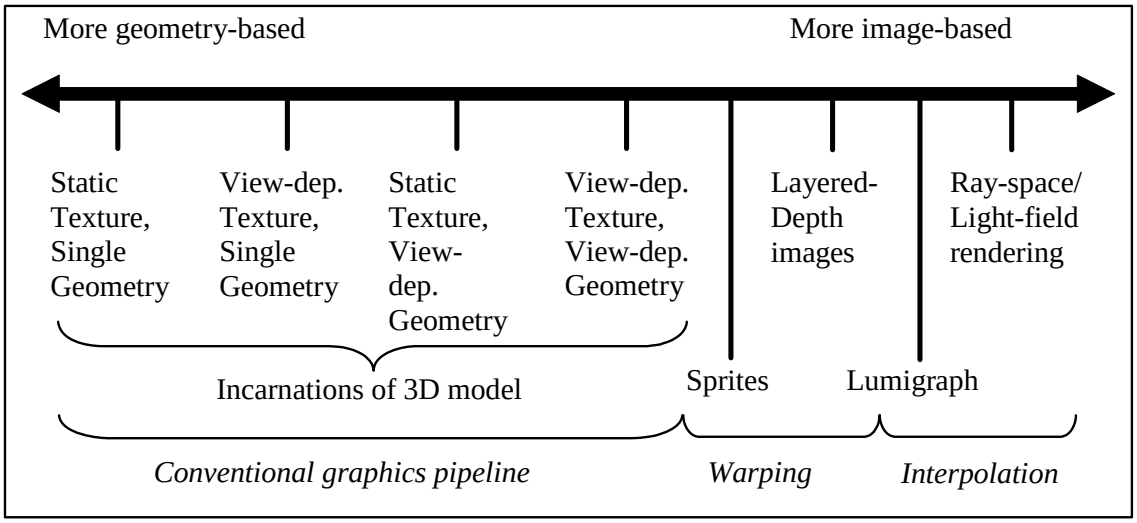

Bild 2: Klassifizierung von Methoden zur 3D Szenenrepräsentation [5]

Im folgenden Abschnitt stellen wir ein FVV System vor, das eine Repräsentation aus Multiview-Video und Tiefenkarten verwendet. Ein ähnlicher Ansatz wird auch in [4] verwendet. Danach folgt die Beschreibung der am Fraunhofer HHI entwickelten Algorithmen für modellbasiertes FVV. Hierbei werden 3D Drahtgittermodelle mit View-dependent Texture Mapping eingesetzt. Ein ähnliches System wird in [15] vorgestellt. Die 3D Geometrie wird mit einem Visual Hull Algorithmus rekonstruiert [7], [8]. In [17] wird gezeigt, wie Vorwissen über die Szene (menschliches Skelettmodell) zu einer verbesserten 3D Rekonstruktion verwendet werden kann. Eine alternative zu 3D Drahtgittermodellen bilden unorganisierte 3D Punktwolken, wie sie z.B. in [16] beschrieben werden.

4. Tiefenbasierte Ansätze

Die tiefenbasierte Zwischenbildsynthese ist einer der klassischen Techniken der 3D-Bildsignalverarbeitung. Im Englischen wird sie auch oft als *Depth Image Based Rendering* (DIBR) bezeichnet. Im Gegensatz zu den modellbasierten Ansätzen, die im nächsten Kapitel ausführlich beschrieben werden, sind DIBR-Methoden meist nicht dazu geeignet, sich mit einer virtuellen Kamera frei im 3D-Raum zu bewegen. In der Regel beschränkt man sich beim DIBR auf eine Navigation entlang der Verbindungslinien zwischen den aufnehmenden Kameras, den sog. Baselines. Man spricht deshalb auch oft von Multi-Baseline-Ansätzen. Sie stehen nicht in Konkurrenz zu den modellbasierten Ansätzen, sondern sind eher komplementär dazu zu sehen, so dass man sie oft, wie in dem evolutionären

Ansatz aus Kapitel 6 beschrieben, geeignet mit modellbasierten Ansätzen kombinieren kann.

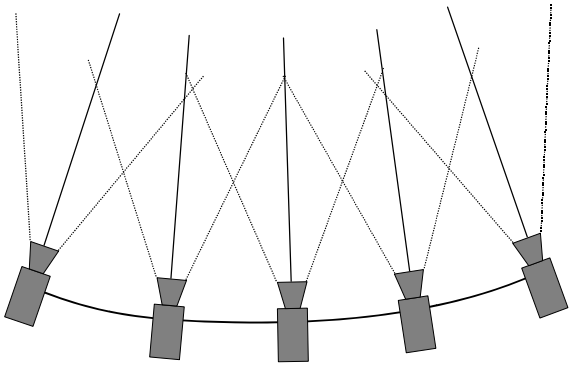

Bild 3: Typische Anordnung eines Multi-Baseline-Systems

Bild 3 zeigt ein Beispiel für einen solchen Multi-Baseline-Ansatz mit fünf Kameras, die auf einer Geraden angeordnet sind. Dabei muss es sich nicht zwangsläufig um eine Anordnung entlang einer Geraden handeln. Die Kameras können sich auch auf einer gekümmten Linie oder einen Kreissegment befinden. Wichtig ist aber, dass sie alle auf einen gemeinsamen Konvergenzpunkt zeigen, also den gleichen Szenenausschnitt von unterschiedlichen Standorten aus aufnehmen. Mit Hilfe einer Tiefenschätzung, wie man sie auch schon aus der klassischen Stereoanalyse kennt, wird nun für jede aufnehmende Kamera eine Tiefenkarte bestimmt. Wie in Bild 4 dargestellt, liefert eine solche Tiefenkarte für jeden Bildpunkt der Kamera einen Tiefenwert, mit dem man auf den dazugehörigen 3D-Raumpunkt zurückschließen kann. Diese Tiefeninformation kann man nutzen, um aus den realen Kamerabildern beliebige virtuelle Ansichten zwischen den Kameras zu interpolieren.

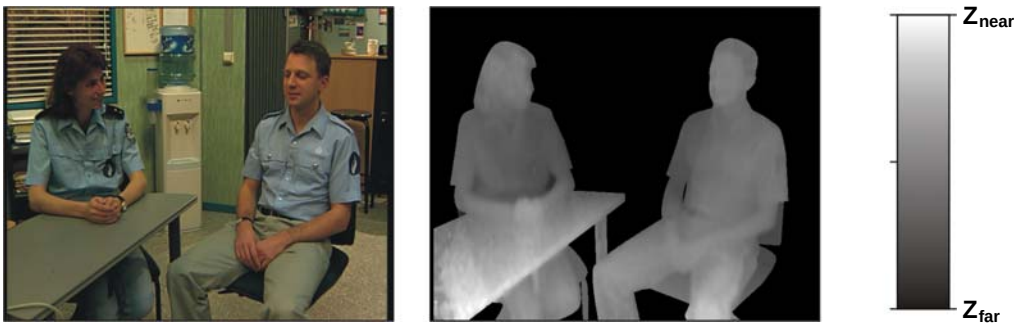

Bild 4: Videobild mit dazugehöriger Tiefenkarte

Die größte Herausforderung eines Multi-Baseline-Ansatzes ist die Schätzung hinreichend genauer Tiefenkarten. Im Folgenden soll ein Beispiel für ein solches Schätzverfahren beschrieben werden. Bild 5 zeigt hierzu eine Testsequenz mit sechs Ansichten, die mit einem Multi-Baseline-System ähnlich zu dem aus Bild 3 aufgenommen wurde. Zu Schätzung der Tiefenkarten, werden unabhängige Stereopaare gebildet (z.B. zwischen benachbarte Kameras). Für jedes Kamerapaar werden dann jeweils zwei Disparitätskarten geschätzt. Eine Disparitätskarte enthält dabei die Punktkorrespondenzen von linker zu rechter Kamera, während die andere Karte die umgekehrte Richtung berücksichtigt. Für die Disparitätsschätzung kann z.B. das block- und pixel-rekursive Verfahren aus [18] verwendet werden.

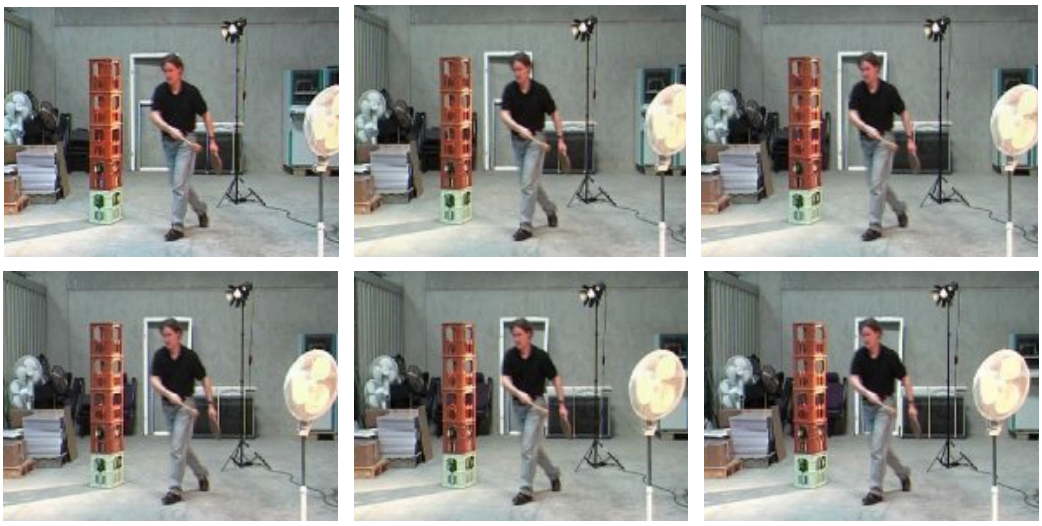

Bild 5: Testsequenz mit sechs Ansichten

Auf diese Weise werden mehrere unabhängige Disparitätskarten erzeugt, die dann gegeneinander auf Konsistenz überprüft werden können. Diese Konsistenzprüfung dient in erster Linie der Fehlererkennung in den geschätzten Disparitätskarten. Bild 6 (links) zeigt hierzu das Ergebnis einer Disparitätsschätzung von der dritten zur vierten Ansicht aus Bild 5 nach der Konsistenzprüfung über mehrere Ansichten. Die schwarzen Pixel beschreiben dabei die Bereiche, deren Disparitätswerte während der Konsistenzprüfung als fehlerhaft erkannt wurden. Die grauen Pixel enthalten dagegen die als konsistent erkannten Disparitätswerte. Üblicherweise entstehen pro Kamera mehrere solcher auf Konsistenz geprüften Disparitätskarten. Sie können dann über Kalibrierungsdaten zu einer gemeinsamen Tiefenkarte zusammengefügt werden. Eventuelle, nicht definierte Bereiche werden dann über segmentgestützte Interpolationstechniken aufgefüllt, die insbesondere auch das Verdeckungsproblem berücksichtigen [19]. Auf diese

Weise entsteht für jede Kamera eine passende Tiefenkarte, die aufgrund der vorangegangenen Verarbeitung konsistent zu den Tiefenkarten der anderen Kameras ist. Das mittlere Bild aus Bild 6 zeigt hierzu die resultierende Tiefenkarte für Ansicht 3 aus Bild 5. Diese Tiefenkarten werden dann dazu genutzt, um die gewünschten Zwischenansichten zu berechnen [20]. Hierzu zeigt Bild 6 (rechts) das Syntheseergebnis einer virtuellen Kamera, die sich zwischen Ansicht 2 und 3 aus Bild 5 befindet.

Bild 6: Ergebnisse der Tiefenschätzung und der tiefenbasierten Bildsynthese

5. Modellbasierte Ansätze (Eisert)

Tiefenbasierte Ansätze zur Bildsynthese erlauben meist nur eine eingeschränkte Navigation innerhalb der Szene. Bedingt durch die Repräsentation durch Tiefenkarten, die Verdeckungen und Aufdeckungen nicht korrekt beschreiben können, werden diese Verfahren vor allem für Zwischenbildsynthese verwendet, bei der die virtuelle Kamera sich in der Nähe der realen Aufnahmepositionen befinden sollte. Für eine vollständig freie Navigation in der Szene werden dagegen komplette dreidimensionale Modelle der Objekte und Personen eingesetzt, die aus beliebigen Richtungen betrachtet werden können. Diese 3D Modelle beschreiben dabei sowohl die Geometrie (durch Punktwolken (Voxel, Splats) oder polygonale Netze) als auch die Farbe (Texturen, Multi-View Textures) der Objekte und müssen für Free Viewpoint Video für jeden Zeitschritt des Videos aus den einzelnen Ansichten berechnet werden.

Für die Rekonstruktion der Modelle werden zunächst Kameras um die zu betrachtende Szene herum platziert. Im Gegensatz zu den tiefenbasierten Ansätzen sind die Kameras meist weiter voneinander entfernt, decken aber Dom-artig eine komplette Hemisphäre um die Objekte ab. Bild 7 zeigt exemplarisch zwei Konfigurationen, die am Fraunhofer HHI installiert wurden.

Bild 7: Multi-view Aufbauten am HHI. Links: Kreisförmige Konfiguration; Rechts: Voller Kameradom mit 16 synchronisierten Kameras

Die Kameraarrays werden dann kalibriert [21], um deren genaue Lage und Abbildungseigenschaften zu bestimmen, die für eine dreidimensionale Geometrie-bestimmung nötig sind. Für die 3D Rekonstruktion können verschiedene Informationen aus den Bildern herangezogen werden. Besonders verbreitet sind *Silhouettenschnittverfahren* [22], die den Umriss eines Objekts oder einer Person benutzen, um ein 3D Modell zu erzeugen. Diese Verfahren haben den Vorteil, das sie sehr robust sind und auf heutigen Computern in Echtzeit realisiert werden können. Dabei nutzt man aus, dass ein Objekt sich immer innerhalb seiner Silhouette befinden muss. Für ein kalibriertes Setup segmentiert man in jedem Kamerabild das Objekt vom Hintergrund. Die Fläche des Objekts definiert dann zusammen mit der Abbildungsgeometrie ein konusförmiges Volumen, innerhalb dessen sich das reale Objekt befinden muss. Da diese Eigenschaft für sämtliche Kameras gilt, muss die Geometrie auch innerhalb des Schnittpunkts aller Volumina liegen, wie es in Bild 8 zu sehen ist.

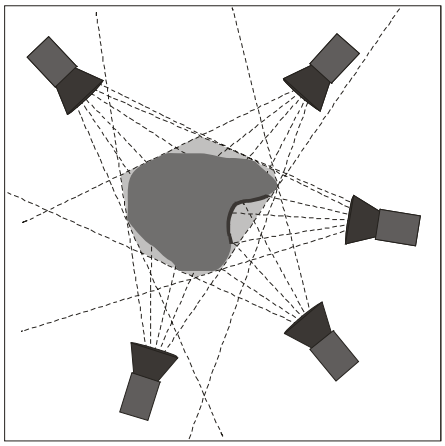

Bild 8: Silhouettenschnittverfahren: Rekonstruktion der konvexen Hülle aus mehreren Kameraansichten

Der Schnitt mit allen Volumina lässt sich dabei sehr effizient mit Volumenbeschreibungen erzielen. Dazu wird das zu rekonstruierende 3D Volumen in kleine Volumenelemente (*Voxel*) unterteilt. Jeder dieser Voxel wird in alle Ansichten projiziert und gelöscht, falls er in mindestens einem Bild außerhalb der Silhouette des Objekts liegt. Es entsteht dabei eine Objektbeschreibung, die aus vielen kleinen Würfeln besteht und die effizient mit einem Octree repräsentiert werden kann.

Bei ausschließlicher Verwendung der Silhouetteninformation lässt sich allerdings nicht die exakte Geometrie sondern nur die *konvexe Hülle* des Objekts rekonstruieren. Kleine Dellen in der Oberfläche, die in der Silhouette nicht sichtbar werden, können auch nicht modelliert werden-. Abhilfe schafft hier die zusätzliche Verwendung von Farbinformation. In Verfahren des *Voxel Colorings* oder *Space Carvings* [23], [24], [25] wird ein photo-konsistentes Objekt rekonstruiert, dessen Rückprojektion in alle Ansichten die richtigen Farben wiedergibt und damit auch nicht-konvexe Objekte rekonstruieren kann. Bild 9 zeigt ein Voxelmodell, das aus einer Kameradomkonfiguration mit 16 Kameras bestimmt wurde.

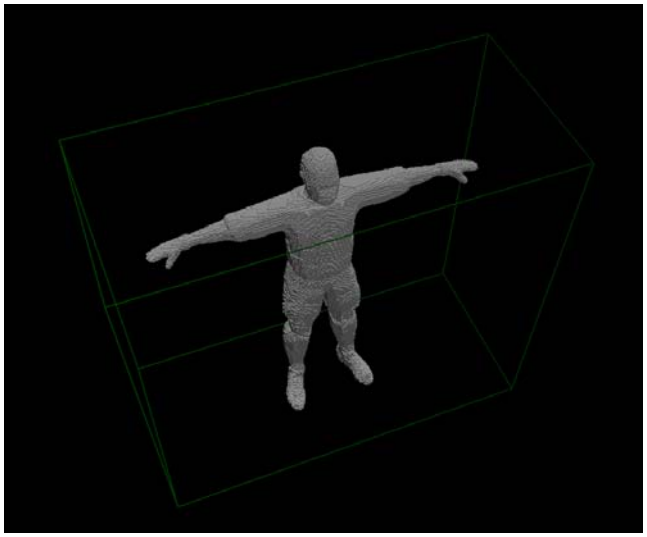

Bild 9: Voxelbasierte Rekonstruktion einer Person im Kameradom

Die Voxelmodelle lassen sich sehr effizient aus den Bilddaten erzeugen. Für eine Speicherung, Übertragung und Darstellung ist allerdings oft eine Oberflächen-basierte polygonale Darstellung besser geeignet, da diese auch hervorragend durch Graphikkarten unterstützt wird. Dazu muss die Volumendarstellung mit Verfahren wie dem *Marching Cubes* [26] in ein Dreiecksnetz übergeführt werden. Die Farbinformation wird meist aus den Kameraansichten extrahiert und in einer Texturkarte gespeichert. Bild 10 zeigt eine Rekonstruktion einer altperuanischen Vase aus 36 Ansichten als Volumenmodell, Drahtgitterdarstellung und in einer texturierten Version.

Bild 10: Links: Voxelmodell, Mitte: Drahtgittermodell, Rechts: Texturiertes Modell einer altperuanischen Vase.

Um blickpunktabhängige Veränderungen der Farbe und Reflexionseigenschaften besser beschreiben zu können, lässt sich statt einer einzigen Textur auch mehrere Texturen verwenden, die dann abhängig von der aktuellen Betrachtungsposition übergeblendet werden können [27].

6. Ein evolutionäres Szenario für FVV

Aufbauend auf die zuvor genannten Techniken soll am HHI ein *FVV*-System entwickelt werden, dass für Live-Events wie Sportveranstaltungen eingesetzt werden kann. Dabei soll es sich um einen interaktiven Service handeln, der zusätzlich zu einer Live-Übertragung angeboten werden kann. Die Interaktivität besteht darin, dass man während der Wiederholung einer besonderen Szene (Replay, Zeitlupe, etc.) den Blickpunkt frei wählen kann und ihn darüber hinaus auch während der Betrachtung der Wiederholung ändern kann. Dies kann zum einen für die Postproduktion interessant sein, da man Wiederholungen von z.B. Torschüssen unter einer frei wählbaren und wechselnden Perspektive mehrmals zeigen kann. In Zukunft wäre es aber auch denkbar, dass diese Funktionalität als neuer Dienst für den Endverbraucher angeboten wird. In diesem Fall würde ein Nutzer eine geeignete *FVV*-Datenrepräsentation der gewünschten Szene von einem Server laden und mit Hilfe eines *FVV*-Players auf seinem Endgerät anschauen. Dabei könnte er, ähnlich zu VR-Applikationen, beliebig mit einer virtuellen Kamera in der Szene navigieren.

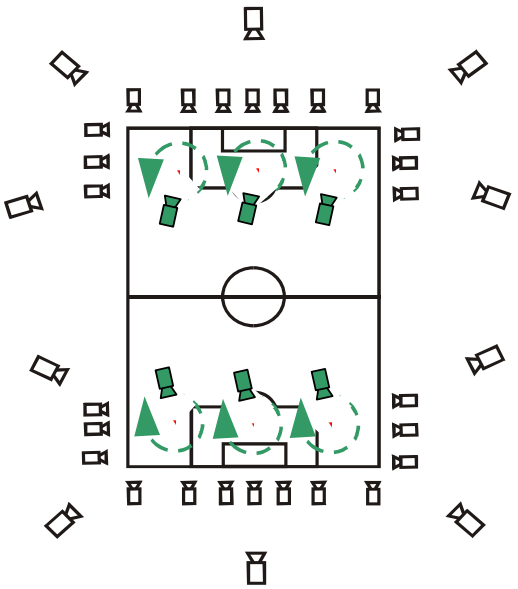

Bild 11: Modulares Mehrkamera-Aufnahmesystem für FVV-Applikationen

Um eine solchen Dienst anbieten zu können, müssen die angebotenen FVV-Szenen entsprechend aufgenommen und aufbereitet werden. Am HHI soll hierzu ein modulares Mehrkamera- und Aufnahmesystem entwickelt werden. Dabei soll insbesondere ein evolutionärer Ansatz verfolgt werden, der mehrere Ausbaustufen bietet und der deshalb auch stufenweise in den Markt eingeführt werden kann. Im einfachsten Fall werden mehrere fest installierte Kameras um ein Ereignis herum gruppiert werden. Wichtig ist dabei, dass alle Kameras auf einen gemeinsamen Konvergenzpunkt ausgerichtet sind (s. rote Markierung in Bild 12 links). Der Nutzer kann zwischen diesen unterschiedlichen Perspektiven wählen oder die Bilder der Kameras hintereinander aufrufen und auf diese Weise eine individuelle, virtuelle Kamerafahrt durchführen. Sofern die Kameras eine höhere Auflösung verwenden als das Ausgangsformat benötigt, kann man durch geeignete geometrische Verzerrung der Bilder auch eine virtuelle Konvergenzpunkt-verschiebung erzielen (s. gelbe Markierungen in Bild 12 links). Auf diese Weise kann man die räumliche Perspektive der virtuellen Kamerafahrt gezielt steuern und verändern.

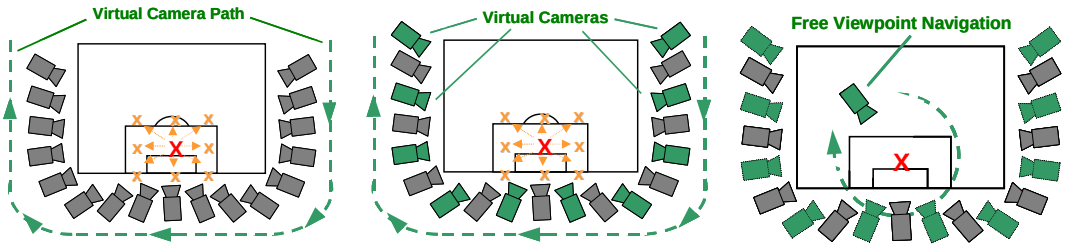

Bild 12: Unterschiedliche Ausbaustufen und Module des FVV-Systems

In einer weiteren Ausbaustufe könnte man dieses System um Techniken zur Tiefenschätzung erweitern. Zu diesem Zweck würde man, wie in Kapitel 4 beschrieben, für jede Kamera eine dazugehörige Tiefenkarte bestimmen. Mit Hilfe dieser Tiefeninformation könnte man dann beliebige Zwischenansichten entlang des virtuellen Kamerapfades berechnen (s. Bild 12 Mitte). Dies hätte zum einen den Vorteil, dass man Kameras einsparen kann. Zum anderen besteht die Möglichkeit, eine nahezu kontinuierlichen Zwischenbildsynthese. Man kann also beliebig viele Ansichten mit einer beliebigen örtlichen Dichte berechnen und damit auch die Geschwindigkeit der virtuellen Kamerafahrt bestimmen und individuell anpassen. Allerdings wäre auch diese zweite Ausbaustufe auf eine virtuelle Fahrt entlang der realen Kameraanordnung gebunden. In einem dritten Schritt sollen deshalb Techniken der modellbasierten Tiefenanalyse hinzugefügt werden (s. Kapitel 5). Dabei können diese Ansätze auf die aus der Vorstufe vorhanden Tiefenkarten aufbauen und diese um Segmentierung und objektbasierte Rekonstruktion von 3D-Szenen und –Objekten erweitern. Diese Erweiterungsstufe würde es dann ermöglichen, dass der Nutzer sich mit einer virtuellen Kamera frei in der rekonstruierten 3D-Szene bewegen kann (s. Bild 12 rechts).

7. Ausblick

Free Viewpoint Video (FVV) stellt eine neue Qualität im Video-Broadcast-Bereich dar. FVV ermöglicht es dem Zuschauer, seine eigene Perspektive beim Betrachten einer Szene zu wählen. Besonders geeignet für ein solches Szenario sind Sportveranstaltungen, aber auch Aufnahmen von Konzerten oder Theaterstücken, wo der Nutzer sich auf bestimmte Szenenausschnitte konzentrieren kann.

Allerdings sind noch vielfältige Entwicklungsschritte notwendig, bevor man dem Zuschauer diese neue Art der Fernsehunterhaltung bieten kann. Speziell in Japan, Korea und USA (Microsoft) hat man die Wichtigkeit des Themas erkannt und dementsprechend auch große Forschungskapazitäten in diesem Bereich aufgebaut. Parallel dazu wurden im Rahmen von ISO-MPEG erste Ansätze für die Standardisierung solcher Systeme gestartet.

Jedoch brauchen sich speziell die deutschen Forschungsgruppen, die sich mit dieser Thematik beschäftigen, nicht hinter ihrer Konkurrenz zu verstecken, was speziell auch durch Leitungsfunktionen in diesem Bereich durch HHI-Mitarbeiter bei ISO-MPEG dokumentiert wird. Allerdings ist es wichtig, dass auch auf politischer Ebene und auf der Ebene der Fernsehproduzenten das Potential dieser Technologie erkannt wird, um den Entwicklungen in Deutschland auch die notwendige Nachhaltigkeit zu verleihen.

8. Literatur

[1] T. Kanade, P. Rander, and P.J. Narayanan, “Virtualized Reality: Constructing Virtual Worlds from Real Scenes”, IEEE MultiMedia, Vol. 4, No.1, jan.-Mar. 1997, pp. 34-37.

[2] A. Smolic, and P. Kauff, “Interactive 3D Video Representation and Coding Technologies”, Proceedings of the IEEE, Special Issue on Advances in Video Coding and Delivery, vol. 93, no. 1, Jan. 2005

[3] Masayuki Tanimoto, “Free Viewpoint Television - FTV”, Proc. PCS 2004, Picture Coding Symposium, San Francisco, CA, USA, December 15.-17. 2004.

[4] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, “High-Quality Video View Interpolation Using a Layered Representation”, SIGGRAPH04, Los Angeles, CA, USA, August 2004.

[5] S.B. Kang, R. Szeliski, and P. Anandan, “The Geometry-Image Representation Tradeoff for Rendering”, Proc. ICIP2000, IEEE International Conference on Image Processing, Vancouver, Canada, September 2000.

[6] M. Levoy, and P. Hanrahan, “Light Field Rendering”, Proc. ACM SIGGRAPH, pp. 31-42, August 1996.

[7] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, and L. McMillan, “Image-Based Visual Hulls”, Proc. SIGGRAPH 2000, pages 369–374, 2000.
[8] Matusik, W., Buehler, C., and McMillan, L., “Polyhedral Visual Hulls for Real-Time Rendering”, Proc. Eurographics Workshop on Rendering 2001.
[9] P. Debevec, C. Taylor, and J. Malik, “Modeling and rendering architecture from photographs: A hybrid geometry- and image based approach”, Proceedings of SIGGRAPH 1996, pp. 11-20, 1996.
[10] H.Y. Shum, and L.W. He, “Rendering with Concentric Mosaics”, Proc. ACM SIGGRAPH, pp. 299-306, August 1999.
[11] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen, “Unstructured Lumigraph Rendering”, Proceedings of SIGGRAPH 2001, pp. 425-432, 2001.
[12] S.J. Gortler, R. Grzesczuk, R. Szeliski, and M.F. Cohen, “The Lumigraph", ACM SIGGRAPH '96, pp.43-54, Aug. 1996.
[13] J. Shade, S. Gortler, L.W. He, and R. Szeliski, “Layered Depth Images”, Proc. SIGGRAPH '98, Orlando, FL, USA, July 1998.
[14] D. Wood, D. Azuma, W. Aldinger, B. Curless, T. Duchamp, D. Salesin, and W. Stuetzle, “Surface Light Fields for 3D Photography”, Proceedings of SIGGRAPH 2000.
[15] T. Matsuyama, X. Wu, T. Takai, and T. Wada, “Real-Time Dynamic 3-D Object Shape Reconstruction and High-Fidelity Texture Mapping for 3-D Video”, IEEE Trans. on Circuits and Systems for Video Technology, Vol. 14, No. 3, pp. 357-369, March 2004.
[16] S. Würmlin, E. Lamboray, and M. Gross, “3D video fragments: dynamic point samples for real-time free-viewpoint video”, Computers and Graphics 28 (1), Special Issue on Coding, Compression and Streaming Techniques for 3D and Multimedia Data, pp. 3-14, Elsevier Ltd, 2004.
[17] J. Carranza, C. Theobalt, M. Magnor, and H.-P. Seidel, “Free-Viewpoint Video of Human Actors”, ACM Trans. on Graphics (special issue SIGGRAPH'03), vol. 22, no. 3, pp. 569-577, July 2003.
[18] N. Atzpadin, P. Kauff, and O. Schreer, “Stereo Analysis by Hybrid Recursive Matching for Real-Time Immersive Video Conferencing”, IEEE Trans. CSVT, Special Issue on Immersive Telecommunications, pp. 321-334, March 2004.
[19] C. Fehn, E. Cooke, O. Schreer and P. Kauff, ”3D Analysis and Image-Based Rendering for Immersive TV Applications”, Signal Processing: Image Communication, 17(2):705-715, October 2002
[20] C. Fehn, N. Atzpadin, M. Müller, O. Schreer, A. Smolic, R. Tanger, P. Kauff. „An Advanced 3DTV System Providing Interoperability and Scalability for a Wide Range Multi-Baseline Geometries”, Proc. of ICIP 2006, Oct. 2006, Atlante, USA.
[21] P. Eisert, "Model-based Camera Calibration Using Analysis by Synthesis Techniques," Proc. 7th International Workshop VISION, MODELING, AND VISUALIZATION 2002, Erlangen, Germany, pp. 307-314, November 2002.

[22] A. Laurentini, “The visual hull concept for silhouette-based image understanding”, IEEE Transactions on Pattern Analysis and Machine Intelligence 16(2), pp. 150-162, 1994.
[23] S. M. Seitz and C. R. Dyer, “Photorealistic scene reconstruction by voxel coloring”, Proc.Computer Vision and Pattern Recognition, pp. 1067-1073, Puerto Rico, 1997.
[24] P. Eisert, E. Steinbach and B. Girod, "Multi-hypothesis, Volumetric Reconstruction of 3-D Objects from Multiple Calibrated Camera Views," International Conference on Acoustics Speech and Signal Processing, ICASSP 99, pp. 3509-3512, Phoenix, March 1999.
[25] K. N. Kutulakos and S. M. Seitz, “A theory of shape by space carving”, International Journal of Computer Vision, 2000.
[26] W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3D surface construction algorithm”, Proc. Computer Graphics (SIGGRAPH), vol. 21, pp. 163-169, 1987.
[27] K. Müller, A. Smolic, P. Merkle, B. Kaspar, P. Eisert, and T. Wiegand, "3D Reconstruction of Natural Scenes with View-Adaptive Multi-Texturing," Proc. 2nd Intl Symp 3D Data Processing, Visualization, and Transmission (3DPVT 2004), Thessaloniki, Greece, pp. 116-123, September 2004.