

Friedrich-Alexander-Universität Erlangen-Nürnberg

**Lehrstuhl für Multimediakommunikation und
Signalverarbeitung**

Prof. Dr.-Ing. André Kaup

Forschungspraktikum

**Auswertung des Einflusses
unterschiedlicher Verfahren zur
Bewegungsschätzung in einem hybriden
Super-Resolution-Ansatz für komprimierte
Videodaten**

von Christoph Schürle

November 2016

Betreuer: Michel Bätz

Erklärung

Ich versichere, dass ich die vorliegende Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe, und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Ort, Datum

Unterschrift

Inhaltsverzeichnis

Kurzfassung	IV
1 Einleitung	1
2 Super Resolution	3
2.1 Single Image Super Resolution	3
2.2 Multi Image Super Resolution	4
2.3 Hybrid Super Resolution	5
3 Motion Estimation	8
3.1 Grundlagen der Bewegungsschätzung	8
3.2 Blockbased Motion Estimation	9
3.3 Wideblock Matching	11
3.4 Parallel Predictive Block Matching	12
3.5 Optical Flow	14
3.5.1 Optical Flow TV-L1	14
3.5.2 Optical Flow Farneback	15
3.6 HEVC	15
4 Simulation	18
4.1 Testsequenzen	18
4.2 Qualitätsmetrik	19
4.3 Versuchsaufbau	20

4.3.1	Block Matching	20
4.3.2	Wideblock Matching	20
4.3.3	Parallel Predictive Block Matching	21
4.3.4	Optical-Flow TV L1	21
4.3.5	Optical-Flow Farneback	21
4.3.6	HEVC	22
4.4	Simulationsergebnisse	22
5	Ausblick	28
	Abbildungsverzeichnis	28
	Tabellenverzeichnis	29

Kurzfassung

In dieser Arbeit wird der Einfluss der Bewegungsschätzer Block Matching, Wideblock Matching, Parallel Predictive Block Matching, OpticalFlow TV-L1, OpticalFlow Farneback und HEVC auf ein Hybrides Super Resolution-Verfahren untersucht und es wird gezeigt, dass mit den beiden Optischen Fluss Verfahren OpticalFlow TV-L1 und OpticalFlow Farneback die besten Resultate erzielt werden können.

Kapitel 1

Einleitung

Bei der Darstellung von Bildern und Videos verlangt der Konsument nach immer höheren Auflösungen, d.h. immer mehr Pixel sollen pro Bildfläche angezeigt werden. Ein praktikabler Weg diese Forderung zu erfüllen, ist es die Pixelgröße bei der Herstellung der eingesetzten Bildsensoren weiter und weiter zu reduzieren. Dies ist jedoch, wie andere Hardwareoptimierungen, oftmals ein teurer Prozess, welcher in teuren Endgeräten für den Verbraucher mündet. Als kostengünstigere Alternative bietet es sich an, mit niedrigauflösender Hardware zu arbeiten und die so gewonnenen Aufnahmen mit Hilfe von Softwarelösungen zu bearbeiten und so höhere Auflösungen zu gewinnen. Ein solches Verfahren ist die Superresolution [?]. Hierbei werden Signalverarbeitungstechniken angewendet, um auf Grundlage eines niedrig aufgelösten Bildes (Single Image Super Resolution = SISR) oder mit Hilfe mehrerer niedrig aufgelöster Frames eines Videos (Multi Image Super Resolution = MISR) eine Erhöhung der Auflösung zu Erzielen. Des Weiteren lohnt es sich einen hybriden Ansatz in der Superresolution zu verwenden, bei dem die Ergebnisse eines SISR-Verfahrens mit denen eines MISR-Verfahrens über eine Entscheidungsmaske kombiniert werden, um das Resultat weiter zu verbessern. Um ein MISR-Verfahren auf eine Anzahl niedrig aufgelöster Frames anwenden zu können, ist es entscheidend die Bewegung von Objekten innerhalb dieses Zeitfensters zu registrieren und zu verfolgen. Dies gelingt durch Anwendung eines Bewegungsschätzers. Es existiert eine Vielzahl von Bewegungsschätzungsverfahren

mit ihren eigenen Vor- und Nachteilen und im Rahmen dieser Arbeit soll untersucht werden, welche Auswirkungen die Auswahl und Parametrisierung eines bestimmten Bewegungsschätzers auf ein hybrides Superresolutionsverfahren hat. Dazu werde ich im Folgenden in Kapitel 2 den Kern der Superresolution, sowie den beispielbasierten SISR Ansatz von Kim et al. [2], den ungleichmäßigen interpolationsbasierten MISR Ansatz von Park et al. [1] und die Zusammenführung dieser beiden Verfahren in einem hybriden Superresolutionsverfahren von Bätz et al. [3] erläutern. In Kapitel 3 stelle ich die untersuchten Bewegungsschätzer, Block-Based Motion Estimation, Wideblock-Matching, Parallel Predictive Block Matching (PPBM), Optical Flow TV-L1, Optical Flow Farneback, und HEVC vor. In Kapitel 4 wird der verwendete Versuchsaufbau zur Untersuchung der Auswirkung verschiedener Bewegungsschätzer im ungleichmäßigen interpolationsbasierten MISR Ansatz aufgezeigt. Anschließend werden in Kapitel 4.4 die aus den Simulationen gewonnenen Ergebnisse analysiert. Abschließend gebe ich einen Ausblick über noch offene Fragestellungen und potentielle zukünftige Forschungen.

Kapitel 2

Super Resolution

Als Super Resolution (SR) bezeichnet man Methoden, deren Ziel die Erhöhung der Auflösung zwei-dimensionaler Bilder ist. Es ist zu unterscheiden zwischen Single Image Super Resolution Verfahren (SISR), welche als Input lediglich ein niedrig aufgelöstes Bild benötigen, und Multi Image Super Resolution (MISR) Verfahren, welche als Input eine Abfolge von Bildern, d.h. eine Videosequenz, benötigen. Eine Kombination aus beiden Verfahren mit Hilfe einer Entscheidungsmaske wird als Hybride Super Resolution (HYSR) bezeichnet. Im Folgenden werden die für diese Arbeit verwendeten Ansätze für SISR, MISR und HYSR erläutert.

2.1 Single Image Super Resolution

Es existieren eine Vielzahl an Strategien für die Single Image Super Resolution (SISR), welche von kanten-gerichteter Interpolation (eng.: edge-directed interpolation) bis hin zu beispelbasierten Methoden (eng.: example-based methods), welche die Korrelation von niedrig aufgelösten zu hoch aufgelösten Bildteilen zu finden versuchen.

Der hier verwendete Ansatz für die Single Image Super Resolution basiert auf der Veröffentlichung von Kim et. al [2]. Die grundlegende Idee dabei ist es mit Hilfe von Beispielpaaren von Eingabe und Ausgabebildern ein Mapping zu erlernen, welches niedrig aufgelöste Eingabebilder auf hochaufgelöste Zielbilder abbildet. Für diesen Zweck wird

die Kernel ridge regression angewendet. Um die zeitliche Komplexität der Trainings- und Testingphase der Kernel ridge regression zu verringern, wird durch die Kombination von kernel matching pursuit und gradient descent eine dünnbesetzte Lösung gefunden. Zusätzlich wird ein a-priori Modell einer allgemeinen Bildklasse herangezogen, welches die diskontinuierlichen Eigenschaften von Bildern berücksichtigt und somit weniger verrauschte Ergebnisse liefert und sowohl Ringingartefakte, als auch die Verschmierung von scharfen Kanten verhindert.

Für die genaue Funktionsweise des Verfahrens siehe [2].

2.2 Multi Image Super Resolution

Auch für Multi Image Super Resolution (MISR) Verfahren lassen sich verschiedene Strategien verfolgen, so finden sich Verfahren, die im Frequenzbereich (eng.: frequency-domain) arbeiten, oder auf deterministischen und stochastischen Regulierungsansätzen (eng.: deterministic and stochastic regularization) beruhen, oder komplett auf eine Bewegungsschätzung verzichten.

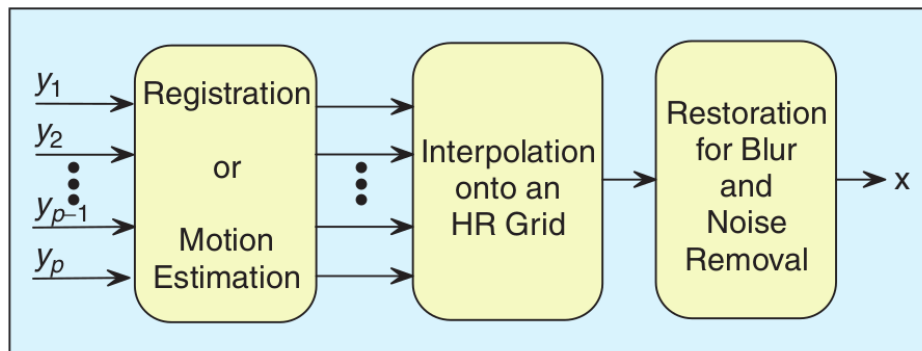


Abbildung 2.1: Ablauf des MISR Verfahrens

Der hier verwendete Ansatz ist ein ungleichmäßiges Interpolationsverfahren (nonuniform interpolation approach) [1], welcher in 3 Stufen abläuft (vgl. Abbildung 2.1):

- Bewegungsschätzung bzw. Registrierung
- Ungleichmäßige Interpolation auf ein Pixelraster mit höherer Auflösung

- Entfernen von Rauschen und Verschmierungen (Deblurring)

Im ersten Schritt werden die relativen Verschiebungen, d.h. die Bewegungsinformation, der Referenzframes zu dem zentralen Frame, welcher hoch aufgelöst werden soll, erfasst. Der zentrale Frame wird dann ungleichmäßig auf die höhere Auflösung interpoliert, wobei hier die Referenzframes mit einfließen, da deren Position im Subpixelbereich des hochauflösenden Bilds nun bekannt ist. (siehe Abbildung 2.2) Abschließend werden auftretendes Rauschen und Weichzeichnungen im hochauflösten Frame mit Hilfe von Entfaltungsverfahren (eng.: deconvolution), welche vorhandenes Rauschen mit einbeziehen, reduziert. Ungleichmäßige Interpolationsverfahren haben eine niedrige Rechenkomplexität, was sie geeignet für Echtzeitanwendungen macht. Die Optimalität des Verfahrens ist jedoch nicht gegeben, da der Rekonstruktionsschritt am Ende die im Interpolationsschritt auftretenden Fehler nicht mit einbezieht.

Für genauere Informationen und Simulationsergebnisse siehe [1].

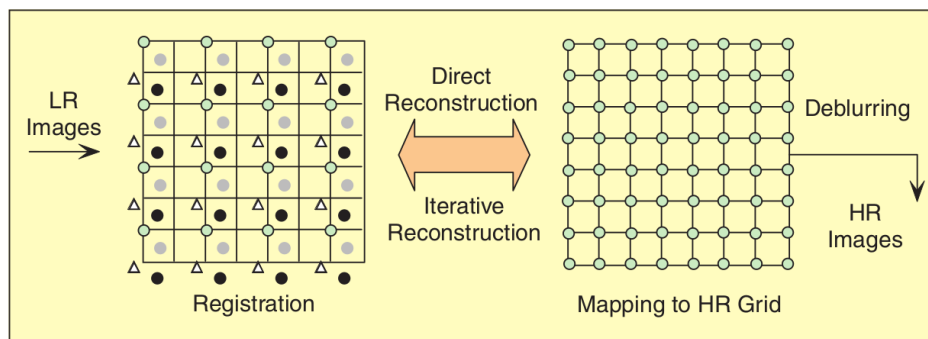


Abbildung 2.2: Rekonstruktionsschritt

2.3 Hybrid Super Resolution

Das in dieser Arbeit untersuchte Hybride Super Resolution Verfahren (HYSR), basiert auf einer Kombination der bereits genannten SISR und MISR Verfahren mit Hilfe einer weichen Entscheidungsmaske. [3]

Der Vorteil besteht darin, dass für Bereiche mit fehlerhafter Bewegungsschätzung, z.B.

statischen Bildhintergründen, die Ergebnisse des SISR Verfahrens eine stärkere Gewichtung erfahren und für korrekte Bewegungsinformation die MISR Ergebnisse ausschlaggebend sind.

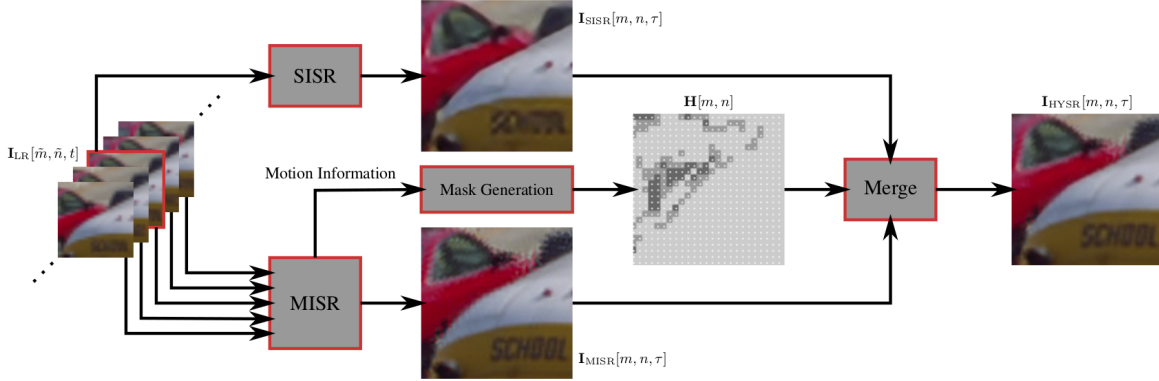


Abbildung 2.3: Hybride Super Resoultion Framework

Bild 2.3 veranschaulicht die funktionsweise der verwendeten HYSR Methode.

Zur Generierung der weichen Entscheidungsmaske werden die vom Bewegungsschätzer der MISR Methode gelieferten Bewegungsvektoren auf ihre Validität überprüft. Dazu wird sowohl ein Vorwärtsvektor (vom Referenzframe zum zentralen, hochaufzulösenden Frame), als auch ein Rückwärtsvektor (vom zentralen, hochaufzulösenden Frame zum Referenzframe) ermittelt. Anschließend findet ein Crosscheck dieser beiden Vektoren statt. Dazu folgt man dem Rückwärtsvektor mit Floating-Point-Präzision im rückwärtsgerichteten Bewegungsfeld an seine Endposition und rundet diese Endposition auf den nächsten Integerwert. Die Differenz zwischen Integer- und Floating-Point-Präzision wird gespeichert. Nun folgt man dem Vorwärtsvektor von der so erhaltenen Integerposition aus im vorwärtsgerichteten Bewegungsfeld und gleicht die Endposition um die gespeicherte Differenz aus. Zeigt der Vektor nun nicht genau auf den Ausgangspunkt, so wird dieser als ungültig in der Entscheidungsmaske H_{init} markiert.

$$H_{init}[m, n] = \sum_{t=1}^{N-1} M_{x_{cu}}[m, n, t]. \quad (2.1)$$

$$H[m, n] = \{ H_{init}[m, n] \forall m, n \in \mathcal{N} \setminus \mathcal{O}, \forall m, n \in \mathcal{N} \cap \mathcal{O}, \text{sonst} \quad (2.2)$$

So kann nun mit Hilfe der Anzahl für jeden Pixel zur Verfügung stehenden, gültigen Bewegungsvektoren die weiche Entscheidungsmaske H nach der Formel 2.2 generiert werden, wobei \mathcal{N} alle Positionen enthält, für die mindestens ein gültiger Bewegungsvektor mit Länge ungleich Null vorliegt. \mathcal{O} enthält alle ursprünglichen Samplepositionen des niedrigaufgelösten Frames, auf welche die Superresolution angewendet wird. Das Ergebnis des HYSR Verfahrens berechnet sich nun nach:

$$I_{HYSR} = (H^2/N^2)I_{MISR} + (1 - H^2/N^2)I_{SISR} \quad (2.3)$$

Kapitel 3

Motion Estimation

Wie in Kapitel 1 beschrieben, spielen Bewegungsschätzer eine entscheidende Rolle in den MISR Verfahren, denn nur durch die Bewegungsschätzung erhält man die essentielle Information der Verschiebung im Subpixelbereich der einzelnen Hilfsframes zu dem zentralen Frame, welcher hochaufgelöst werden soll.

Die Zielsetzung dieser Arbeit ist es nun zu ermitteln, wie sich der Einsatz unterschiedlicher Bewegungsschätzer in einem HYSR-Verfahren auswirkt und ob sich gegebenenfalls ein zu bevorzugender Bewegungsschätzer finden lässt.

Um eine fundierte Aussage über die Eignung eines speziellen Bewegungsschätzers in einem hybriden SR-Verfahren treffen zu können, gilt es zuerst die Grundlagen der Bewegungsschätzung, sowie die Funktionsweise der untersuchten Bewegungsschätzer zu verstehen. Daher werden diese in den folgenden Unterkapiteln näher erläutert.

3.1 Grundlagen der Bewegungsschätzung

Die Bewegungsschätzung, häufig auch als Optischer Fluss (eng.: Optical Flow) bezeichnet, stellt eine Repräsentation der Bewegung innerhalb eines Videos dar. Ein Video besteht aus einer zeitlichen Abfolge von 2D Bildern, sogenannter Frames. Zwischen einzelnen Frames findet meist eine Veränderung des 2D Bildes statt, hervorgerufen z.B. durch Bewegung der Objekte in der Szene, Helligkeitsveränderungen, Bewegung

der Kamera oder ähnliches. Die Aufgabe der Bewegungsschätzer ist es nun die Bewegungsinformation, also die Transformation von einem Bild zum Anderen, zwischen zwei, zeitlich nah beieinander liegender Frames zu ermitteln. Diese Bewegungsinformation, dargestellt als Bewegungsvektoren, kann sich sowohl auf das gesamte Bild, beliebig geformte Teilregionen, rechteckige Teilblöcke, oder jeden einzelnen Pixel des Bilds beziehen. Bewegungsschätzer lassen sich in zwei Kategorien unterteilen, indirekte Bewegungsschätzer, welche feature basiert sind und direkte Bewegungsschätzer, welche hier vorwiegend betrachtet werden und pixelbasiert sind. Indirekte Bewegungsschätzer finden überwiegend Anwendung in der Videocodierung, da diese daran interessiert sind die wahre Bewegung von Objekten zu ermitteln und zu codieren. Bei der direkten Bewegungsschätzung hingegen ist es nicht oberste Priorität die wahre Bewegung von Objekten zu ermitteln, sondern wichtiger ist es die minimale Änderung zwischen dem aktuellen Frame und einem Referenzframe zu finden, um diese anschliessend effizient codieren und übertragen zu können. [4]

3.2 Blockbased Motion Estimation

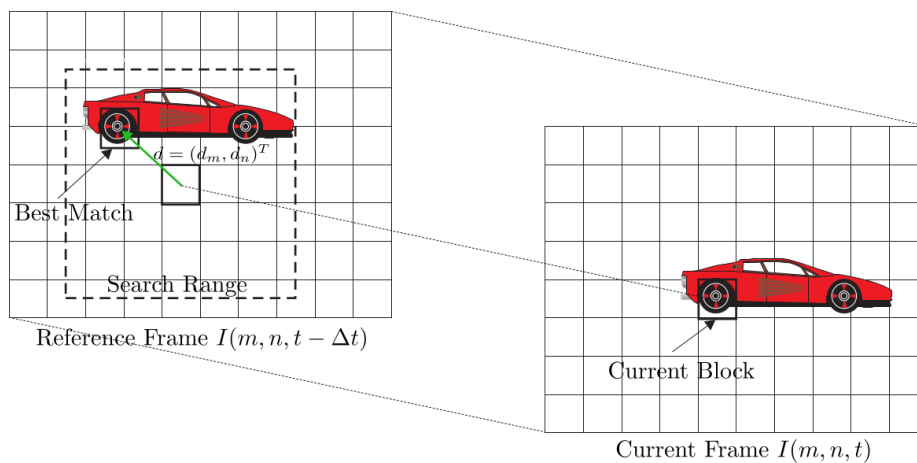


Abbildung 3.1: Fullsearch Block-Matching Algorithmus

Bei den Block-Matching Algorithmen handelt es sich um einen robusten und einfach zu implementierenden Bewegungsschätzer. [4]

Zuerst wird ein Frame in eine bestimmte Anzahl, sich nicht überlappender, gleich großer Macroblocks (rechteckige Teilregionen) unterteilt. Anschließend wird jeder Macroblock im aktuellen Frame mit den Macroblocks in einem definierten Suchraum, im Referenzframe (meist der vorhergegangene oder folgende Frame) verglichen.

Innerhalb dieses Suchraums wird nun nach einem Macroblock mit der größten Übereinstimmung gesucht, wobei die größte Übereinstimmung definiert ist durch das Minimum einer Kostenfunktion, wie z.B. im Falle dieser Arbeit der Sum-of-Squared Differences (SSD):

$$SSD(d_m, d_n) = \sum_{(m,n) \in \text{Block}} (I(m, n, t) - I(m + d_m, n + d_n, t - \Delta t))^2 \quad (3.1)$$

Für jede mögliche Position im Suchraum wird auf diese Weise ein Bewegungsvektor ermittelt und ein SSD-Wert zugeordnet. Sind alle Positionen abgearbeitet wird der Bewegungsvektor mit dem niedrigsten SSD-Wert als Bewegung zwischen dem aktuell betrachteten Macroblock und dem entsprechenden Macroblock im Referenzframe abgespeichert. So wird für jeden Macroblock im aktuell betrachteten Frame genau ein Bewegungsvektor ermittelt. Der gleiche Bewegungsvektor wird allen Pixeln in einem Macroblock zugeteilt.

Die Komplexität und der Rechenaufwand des Bewegungsschätzers sind abhängig von den verwendeten Parametereinstellungen, der Blockgröße, der Searchrange (welche den Suchraum im Referenzframe definiert) und der Step size (die Genauigkeit im Pixel- oder Subpixelbereich). Die Blockgröße bestimmt die Auflösung des resultierenden Bewegungsvektorfelds, wobei durch eine kleine Blockgrößen (d.h. hohe Auflösung) die Bewegungsschätzung fehleranfällig für Rauschen wird. Bei hohen Blockgrößen ist die Auflösung geringer, dafür ist die Bewegungsschätzung stabiler. Die Searchrange bestimmt direkt den maximalen Bewegungsvektor, was zur Folge haben kann, dass bei zu kleiner Searchrange keine großen Bewegungen mehr erfasst werden können. Daher kann auch die wahre Bewegung von Objekten nicht korrekt erfasst werden.

Ein Vorteil von Fullsearch Block-Matching Verfahren ist, dass sie, bei optimal gewählter

Searchrange, garantiert den besten Bewegungsvektor finden. Der Nachteil ist jedoch, dass viele Rechenoperationen benötigt werden und das Verfahren daher langsam und rechenaufwendig ist.

3.3 Wideblock Matching

Beim Wideblock-Matching handelt es sich ebenfalls um ein Block-Matching Verfahren, wie in Kapitel 3.2 beschrieben. [4]

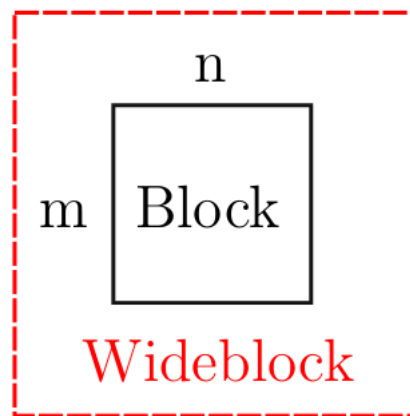


Abbildung 3.2: Aufbau eines Wideblocks

Die Idee ist hier jedoch eine temporär erhöhte Blockgröße zu verwenden, um die beste Entsprechung im Referenzframe zu finden, wobei das Ergebnis nur für die Pixel innerhalb des Blocks der ursprünglichen Größe zugewiesen wird. Dieses Verfahren liefert somit einen guten Kompromiss, zwischen der Auflösung des Bewegungsvektorfelds und der Stabilität der Kostenfunktion, da für größere Blöcke die Berechnung weniger anfällig für Rauschen ist.

3.4 Parallel Predictive Block Matching

Bei dem Parallel Predictive Block Matching (PPBM) Verfahren [4] handelt es sich um einen prediktiven Bewegungsschätzer, d.h. daß zusätzlich zu dem verwendeten Referenzframe auch Informationen aus vorhergegangenen, bereits geschätzten Frames und aus bereits geschätzten Pixeln innerhalb des aktuellen Frames mit einbezogen werden. Es wird die Annahme getroffen, daß Objekte wesentlich größer als die Blockgröße sind und somit die Wahrscheinlichkeit hoch ist, dass die Bewegungsvektoren zweier benachbarter Blöcke sich ähnlich sind. Die zweite getroffene Annahme ist, daß sich die Bewegung der Objekte über der Zeit nur langsam ändert und daher die Bewegungsvektoren zweier zeitlich aufeinander folgender Frames korreliert sind.

Um also einen Bewegungsvektor für einen bestimmten Block zu bestimmen, werden zusätzliche zeitliche und örtliche Prädiktoren benutzt. Für die Position der örtlichen Prädiktoren kommen nur bereits geschätzte Blöcke in Frage, d.h. aufgrund von Kausalität lediglich Blöcke die sich oberhalb oder links des aktuellen Blocks befinden. Die Wahl der Position eines zeitlichen Prädiktors ist völlig frei, er muß lediglich in einem vorher gegangenen Frame liegen. In dem verwendeten PPBM Verfahren werden zwei örtliche und zwei zeitliche Prädiktoren verwendet, wobei jeweils ein paar aus örtlich und zeitlichem Prädiktor einen Schätzer für entweder die horizontale oder vertikale Bewegung bildet.

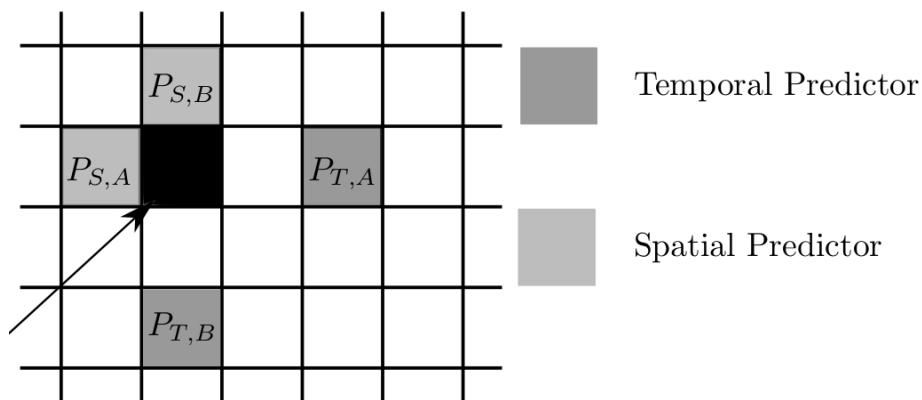


Abbildung 3.3: Position der zeitlichen und örtlichen Prädiktoren in PPBM

In dem hier verwendeten Verfahren, werden neben dem gerade beschriebenen grundlegenden Algorithmus zusätzliche Erweiterungen angewendet:

- Ein abgeänderter Plausibilitätscheck für die Auswahl des Schätzers von vertikaler oder horizontaler Bewegung
- Ein Multi-Reference Frame Ansatz, welcher nicht nur einen direkt zeitlich benachbarten Referenzframe, sondern zusätzlich auch die Information aus zeitlich weiter entfernten Referenzframes mit einbezieht
- Der Einsatz von Optical Flow, um die Qualität der Bewegungsschätzung zu erhöhen
- Hierarchical Wideblock-Splitting, welches iterativ die Auflösung des Bewegungsvektorfeldes anpasst, um die Bewegungsschätzung zu verbessern

Für eine genauere Erläuterungen der oben genannten Erweiterungen siehe [4].

Im Gegensatz zu [4] wird in dieser Arbeit eine sternförmige Bewegungsschätzung, wie in Abbildung 3.4, durchgeführt. Dazu erhält der PPBM-Bewegungsschätzer als Eingabe den Referenz-Frame und den zentralen Frame und die Bewegungsschätzung wird so lange durchgeführt, bis eine Konvergenz der resultierenden Bewegungsvektoren erreicht ist.

Dies vermindert jedoch den Einfluss der temporären Prädiktoren, da lediglich ein Referenzframe zur Verfügung steht. In der Variante von [4] startet die Bewegungsschätzung mit Frame 1 und arbeitet sequentiell bis zum zentralen Frame 5.

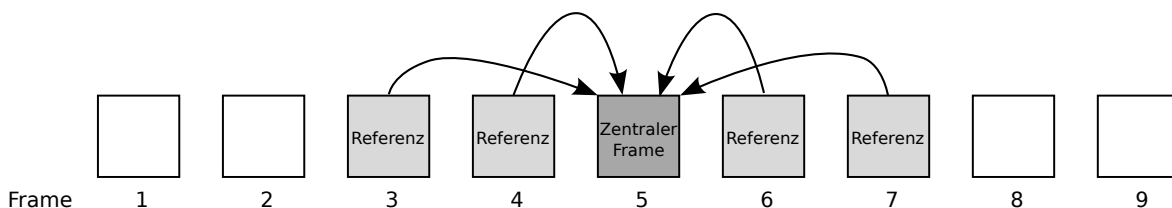


Abbildung 3.4: Sternförmige Bewegungsschätzung

Um den Einfluss zusätzlicher Frames in der sternförmigen Bewegungsschätzung zu untersuchen, werden ebenfalls Simulationen mit einem Pre-PPBM-Frame (siehe Abbildung 3.5) durchgeführt. Dazu wird ein zusätzlicher Frame, im gleichen temporalen Abstand wie der erste Referenzframe zum zentralen Frame, in den PPBM-Bewegungsschätzer gespeißt. Sollte sich der Pre-PPBM-Frame außerhalb der Sequenz befinden wird darauf verzichtet und wie in Abbildung 3.4 verfahren.

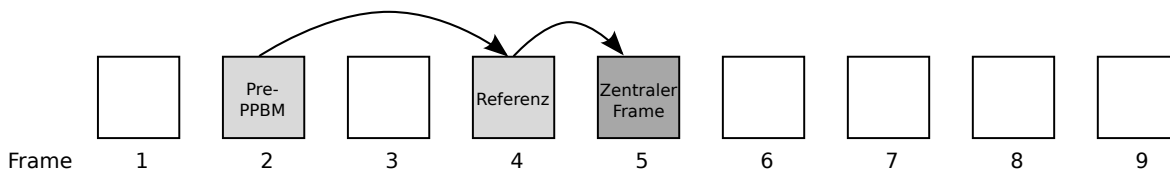


Abbildung 3.5: Sternförmige Bewegungsschätzung mit Pre-PPBM Frame

3.5 Optical Flow

Der Begriff Optischer Fluss (eng.: Optical Flow) beschreibt sowohl die Bewegung innerhalb einer Bildsequenz, als auch die Verfahren, die verwendet werden, um dieses Ergebnis zu erhalten.

Um den Optischen Fluss lokal zu ermitteln werden innerhalb einer definierten Umgebung Helligkeitsmuster betrachtet und mit Hilfe deren Gradienten kann ein Bewegungsvektor bestimmt werden. Dies führt jedoch zu einer Gleichung mit zwei Unbekannten, welche für sich nicht gelöst werden kann. Dies wird auch als Aperturproblem des Optischen Flusses bezeichnet. Um nun den Optischen Fluss zu bestimmen werden in jedem Verfahren zur Ermittlung des Optischen Flusses zusätzliche Einschränkungen eingeführt, damit das Gleichungssystem lösbar wird.

3.5.1 Optical Flow TV-L1

Das OpticalFlow TV L1 Verfahren [5] ist eine differentielle Methode zur Bestimmung des optischen Flusses zwischen zwei Frames.

Der Ansatz basiert auf einer Regulierung der absoluten Variation (eng: total variation

(TV)) und der Anwendung der robusten L1-Norm auf den Datenterm. Der Hauptvorteil dieser Formulierung, gegenüber dem klassischen Ansatz von Horn und Schunck [6], besteht darin, dass Diskontinuitäten im optischen Fluss zugelassen werden und der Ansatz gleichzeitig, im Vergleich zum klassischen Optischenfluss Verfahren, robuster gegenüber Beleuchtungsänderungen, Verdeckungen und Rauschen ist.

Für genaue Informationen und Implementierungsdetails siehe [6] und [5].

3.5.2 Optical Flow Farneback

Beim Farneback Algorithmus [7] werden im ersten Schritt die jeweilige Nachbarschaft aller Pixel der beiden Frames durch quadratische Polynome approximiert. Dies lässt sich effizient durch die Polynomial Expansion Transform realisieren. Die Darstellung durch quadratische Polynome erlaubt es nun eine Verschiebungsmatrix aus den Koeffizienten der Polynomial Expansion Transform abzuleiten, welche mit Hilfe von a-priori Wissen über die Verschiebungsmatrix noch weiter präzisiert werden kann. Diese Verschiebungsmatrix ist nichts anderes als die Bewegung zwischen den beiden betrachteten Frames.

Für genauere Informationen siehe [7].

3.6 HEVC / H.265

Beim High Efficiency Video Coding (HEVC) [4] oder auch H.265 handelt es sich um einen weitverbreiteten Standard zum Kodieren von Videoinhalten. HEVC ist ein hybrider Videocoder, der sich zur Bewegungsschätzung eines Block-Matching Verfahrens (vgl. Kapitel 3.2) bedient, welches durch Quadtree Partitionierung ergänzt wird. Beim HEVC wird nach drei, funktionell unterschiedlichen, Einheiten differenziert. Die coding unit (CU) ist eine Region mit dem gleichen prediction mode (intra oder inter codiert). Die prediction unit (PU) umfasst eine Region, die die gleiche Schätzungsinformation (prediction information) enthält. Die transform unit (TU) enthält die Regionen, die die gleiche Transformation besitzen. Jeder Frame wird in coding tree blocks (CTBs),

d.h. gleich große, sich nicht überlappende Blöcke, unterteilt. Die Blöckgröße lässt sich dabei über die Encodereinstellungen zwischen 16x16, 32x32 oder 64x64 Pixeln wählen. Auf diesen CTBs findet nun die Partitionierung statt, dabei kann ein CTB rekursiv in kleinere Quadtree-Regionen, die coding units (CU), unterteilt werden. Die so gewonnenen CUs sind die Basis für die Schätzung im HEVC. Jede CU kann weiter in prediction units (PUs) und einen Baum von transformation units (TUs) unterteilt werden. Die PUs werden nun für die Bewegungsschätzung verwendet. Dabei wird eine PU mit ihren örtlich benachbarten PUs im Referenzframe verglichen und diese PU ausgewählt, die der betrachteten am ähnlichsten ist. Die Verschiebung zwischen den beiden ist wiederum der Bewegungsvektor. Der Vorteil des HEVC ist dabei, dass er Ergebnisse im Subpixelbereich liefert und effizienter als andere Videocodierer arbeitet.

Für diese Arbeit werden die gewonnenen Bewegungsvektoren aus dem HEVC extrahiert und für eine sternförmige Bewegungsschätzung (siehe Abbildung 3.4) angepasst.

Es existieren hierbei für die Anpassung 3 mögliche Fälle, abhängig davon, ob für einen untersuchten Referenzframe Stützframes zur Verfügung stehen oder nicht. Der HEVC liefert Informationen zur Bewegungsschätzung von einem Referenzframe zu einem Stützframe, jedoch nicht zwingend zum zentralen Frame.

1. Der Referenzframe hat keinen Stützframe.

Es wird eine Block Matching Bewegungsschätzung zum zentralen Frame durchgeführt.

2. Der Referenzframe hat einen Stützframe (zeitlich vorhergehend oder nachfolgend).

Es wird eine lineare Bewegung angenommen, d.h. der Bewegungsvektor zum Stützframe wird mit seiner Länge normiert. Der so normierte Vektor beschreibt die lineare Bewegung zwischen zwei Frames und wird nun mit dem zeitlichen Abstand zum zentralen Frame multipliziert, um die gesuchte Bewegung zu erhalten.

3. Der Referenzframe hat zwei Stützframes, einen zeitlich vorhergehenden und einen

zeitlich nachfolgenden.

Es wird eine lineare Bewegung angenommen. Für beide Stützframes wird der Bewegungsvektor mit seiner Länge normiert und die beiden miteinander verrechnet (weiter entfernte Stützframes fallen weniger ins Gewicht als nahe), um so die geschätzte, normierte Bewegung zwischen zwei benachbarten Frames zu erhalten. Diese Bewegung wird nun mit dem zeitlichen Abstand zum zentralen Frame multipliziert, um die gesuchte Bewegung zu erhalten.

Kapitel 4

Simulation

Im Rahmen dieser Arbeit werden, die in dem Kapitel 3 bereits vorgestellten, Bewegungsschätzungsverfahren

- Block Matching
- Wideblock Matching
- Parallel Predictive Block Matching
- Optical-Flow TV L1
- Optical-Flow Farneback
- HEVC

auf ihre Eignung in dem in Kapitel 2.3 vorgestellten HYSR Ansatz untersucht.

4.1 Testsequenzen

Dazu wurden die folgenden Testsequenzen untersucht:

Mall In der Sequenz ist eine Shoppingallee zu sehen, in welcher sich Passanten in verschiedene Richtungen bewegen während sich die Kamera parallel zur Szenerie bewegt. Die Auflösung ist 832x480 Pixel.

Panslow In dieser Sequenz fährt die Kamera langsam über eine Reihe diverser, stationärer Objekte mit hohem Kontrast. Die Kamerabewegung ist langsam und parallel zur Szenerie. Die Auflösung ist 1280x720 Pixel.

Spincalender Die Sequenz enthält eine langsame Rotationsbewegung der Kamera und zeigt einen Kalender und periodische Muster. Die Auflösung ist 1280x720 Pixel.

Terrace In dieser Sequenz ist eine langsame, diagonale Kamerafahrt zu sehen. Die Kamera fährt beginnt auf der Terasse eines Restaurants und endet auf einer langsam befahrenen Straße. Die Auflösung ist 1920x1080 Pixel.

Kimono Diese Testsequenz enthält einen Kameraschnitt. In der ersten Hälfte ist eine gehende Frau im Profil zu sehen und die Kamera bewegt sich parallel zu ihr in gleicher Geschwindigkeit. In der zweiten Hälfte ist die Frau vor einem Haus zu sehen und eine leichte parallele Kamerabewegung vorhanden. Die Auflösung ist 1920x1080 Pixel.

Traffic Diese Sequenz zeigt eine Autobahn mit langsam und gleichmäßig fahrenden Fahrzeugen, die sich in gerader Linie bewegen. Die Kamera steht still. Die Auflösung ist 2560x1600 Pixel.

4.2 Qualitätsmetrik

Um die Qualität einer Bewegungsschätzung beurteilen zu können wird der Peak Signal-To-Noise Ratio (PSNR) Wert herangezogen. Dieser ist eine gängige Metrik in der Bild- und Videoverarbeitung. Der PSNR Wert ist abhängig vom mean square error (MSE) wie Formeln 4.1 und 4.2 zeigen:

$$\text{MSE} = \frac{1}{M * N} \sum_m \sum_n (I(m, n, t) - \tilde{I}(m, n, t))^2 \quad (4.1)$$

$$\text{PSNR(dB)} = 10 \log \frac{m^2}{\text{MSE}} \quad (4.2)$$

\tilde{I} ist dabei das geschätzte Bild und I das Original.

Ein Nachteil bei der Verwendung von PSNR Werten als Qualitätsmerkmal ist, dass oft die wahrgenommene Qualität von den PSNR Werten abweicht, da PSNR Werte auf einem Pixel für Pixel Vergleich basieren, die menschliche Wahrnehmung jedoch nicht.

4.3 Versuchsaufbau

Jeder untersuchte Bewegungsschätzer aus Kapitel 3 wird für jeweils 5 Frames jeder Testsequenzen aus Kapitel 4.1 in einem HYSR getestet.

Es werden jeweils 3, 7 und 15 Referenzframes in den MISR Teil mit einbezogen.

Die Auflösungserhöhung beträgt den Faktor 4.

Zusätzlich werden folgende Parametereinstellungen der Bewegungsschätzer getestet:

4.3.1 Block Matching

<i>Setup</i>	Blockgröße	Suchweite
<i>1</i>	16	64
<i>2</i>	16	32
<i>3</i>	8	64

Tabelle 4.1: Simulierte Parametersets für Block Matching

4.3.2 Wideblock Matching

<i>Setup</i>	Blockgröße	Suchweite	Margin
<i>1</i>	16	64	8
<i>2</i>	16	32	8
<i>3</i>	8	64	4

Tabelle 4.2: Simulierte Parametersets für Wideblock Matching

4.3.3 Parallel Predictive Block Matching

Die Parameter wurden auf Grundlage von [4] ausgewählt.

<i>Setup</i>	Blockgröße	Pre-PPBM-Frame	Erweiterung
1	16	nein	
2	16	nein	Optical Flow & Wideblock-Splitting
3	16	ja	
4	16	ja	Optical Flow & Wideblock-Splitting

Tabelle 4.3: Simulierte Parametersets für Parallel Predictive Block Matching

4.3.4 Optical-Flow TV L1

Die Parameter wurden auf Grundlage von [9] ausgewählt.

<i>Setup</i>	lambda
1	0,15
2	0,3
3	0,4

Tabelle 4.4: Simulierte Parametersets für Optical Flow TV-L1

tau	theta	n scales	warps	epsilon	iterations
0.25	0.3	5	5	0.001	300

Tabelle 4.5: Konstante Parameter jeder Simulation für Optical Flow TV-L1

4.3.5 Optical-Flow Farneback

Die Parameter wurden auf Grundlage von [8] ausgewählt.

<i>Setup</i>	<i>poly_n</i>	<i>poly_sigma</i>
<i>1</i>	5	1,2
<i>2</i>	7	1,5

Tabelle 4.6: Simulierte Parametersets für Optical Flow Farneback

<i>pyr_scale</i>	<i>levels</i>	<i>iterations</i>
0.5	3	3

Tabelle 4.7: Konstante Parameter jeder Simulation für Optical Flow Farneback

4.3.6 HEVC

Der Quantisierungsfaktor qp wurde zu 0 gewählt, damit die Quantisierung sehr gering ist und somit eine Vergleichbarkeit zu den anderen Bewegungsschätzern gegeben bleibt.

<i>Setup</i>	Suchweite
<i>1</i>	32
<i>2</i>	64

Tabelle 4.8: Simulierte Parametersets für HEVC

4.4 Simulationsergebnisse

Abbildung 4.1 zeigt den mittleren PSNR-Gewinn in dB des HYSR Verfahrens gegenüber einer bikubischen Interpolation, abhängig von der Anzahl der verwendeten Referenzframes. Die Mittelung findet dabei über alle Parametersets und jeweils 5 Frames aller Testsequenzen statt.

Es ist deutlich zu erkennen, dass sowohl der PPBM-Bewegungsschätzer, als auch die Informationen aus der HEVC Bewegungsschätzung nicht geeignet für die Anwendung in einem HYSR Verfahren sind (vgl. Abbildungen 4.9 und 4.10), da sie im PSNR-Vergleich sogar schlechter abschneiden, als die bikubische Interpolation.

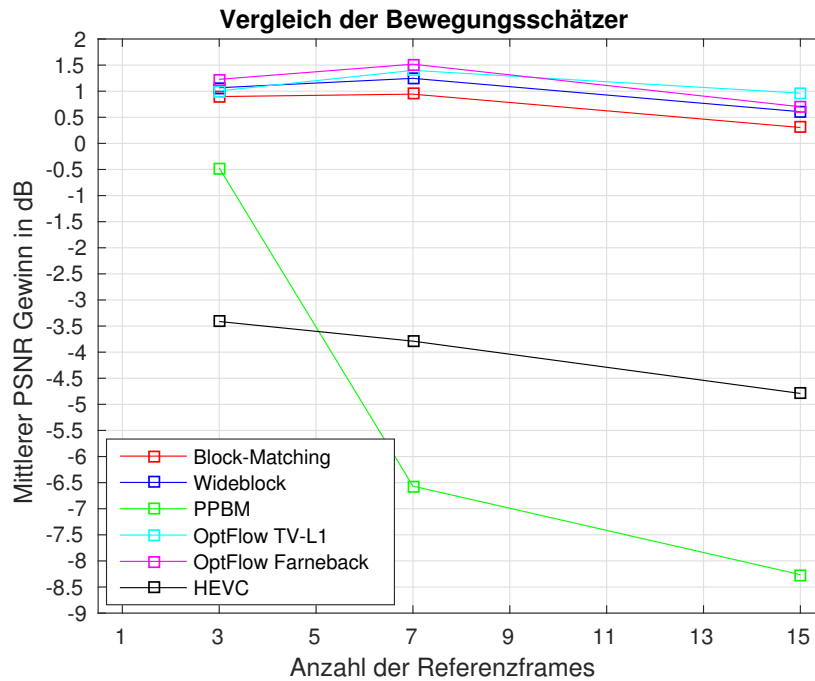


Abbildung 4.1: Simulationsergebnisse aller Bewegungsschätzer

Für den PPBM lässt sich dies durch die Tatsache erklären, dass das Verfahren nicht dazu ausgelegt ist eine sternförmige Bewegungsschätzung (d.h. von jedem Referenzframe direkt zum zentralen SR-Frame und umgekehrt) zu erzeugen. Der Vorteil des PPBM Verfahrens liegt in der Ausnutzung der temporären Prädiktoren, diese sind bei einer sternförmigen Bewegungsschätzung jedoch nur bedingt relevant. Auch das hinzunehmen des Pre-PPBM-Frames in Setup 3 & 4 aus Tabelle 4.3 zeigt keine deutliche Verbesserung (vgl. Abbildung 4.2).

Bei der Ausnutzung der Bewegungsinformationen aus dem HEVC wird eine lineare Bewegung angenommen um eine sternförmige Bewegungsschätzung zu ermöglichen (vgl. Kapitel 3.6). Gerade jedoch wenn für einen Frame nur zeitlich weit entfernte Frames zur Verfügung stehen, ist eine lineare Bewegungsschätzung fehlerhaft, was zu einer starken PSNR-Abnahme führt.

Abbildung 4.3 zeigt, dass für ein HYSR Verfahren, vor Allem für eine höhere Anzahl für Referenzframes, die Optischen Fluss Verfahren die höchsten PSNR-Gewinne erzielen.

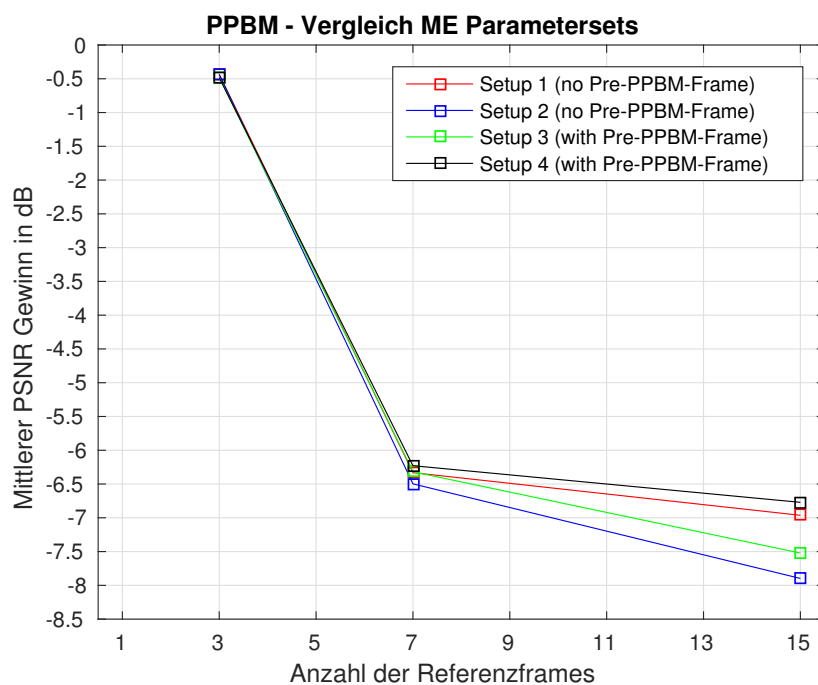


Abbildung 4.2: Vergleich der Parametersets für den PPBM gemittelt über alle Testsequenzen.

Für den optischen Fluss Farneback haben die Simulationen ergeben, dass die untersuchten Parametersets sich nur vernachlässigbar unterscheiden.

In Abbildung 4.4 ist zu erkennen, dass für 3 und 7 Referenzframes der Optische Fluss Farneback den Optischen Fluss TV-L1 Bewegungschätzer überlegen ist. Für 15 Referenzframes ist jedoch der Optische Fluss TV-L1 zu bevorzugen.

Dies legt nahe ein Verfahren für die HYSR zu benutzen, in welchem bis zu 9 Referenzframes (geschätzter Peak, vgl. [3]) der Optische Fluss Farneback und für eine höhere Anzahl an Referenzframes der Optische Fluss TV-L1 mit einem alpha von 0,03 verwendet wird, um den PSNR Gewinn zu maximieren.

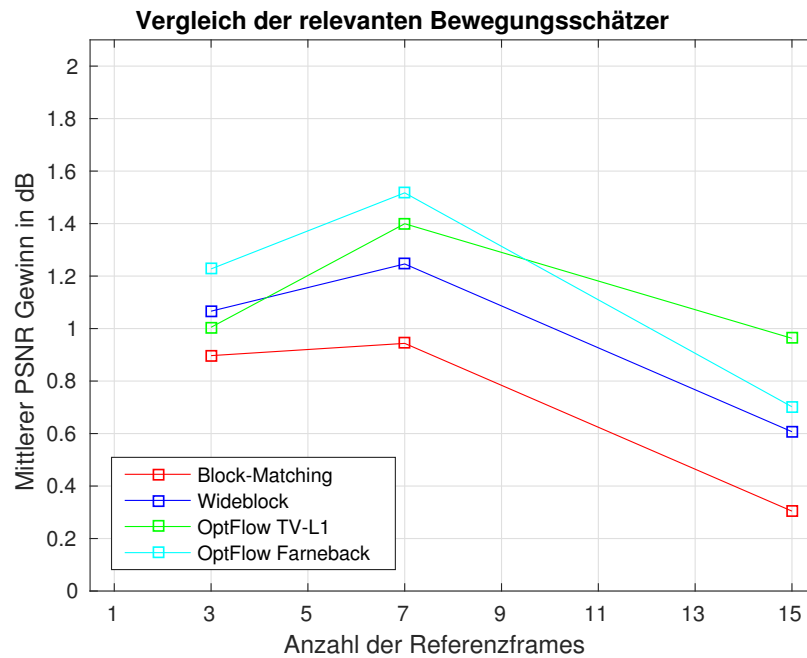


Abbildung 4.3: Vergleich der Bewegungsschätzer mit positivem PSNR-Gewinn gegenüber bikubischer Interpolation. Gemittelt über alle Testsequenzen und Parametersets.

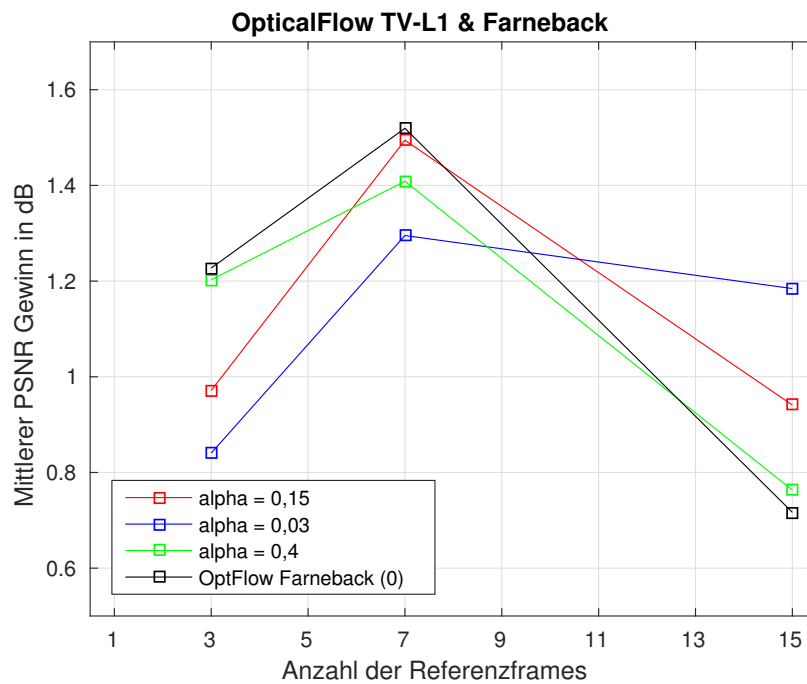


Abbildung 4.4: Vergleich der Parametersets für den Optischen Fluss TV-L1 mit dem Optischen Fluss Farneback, gemittelt über alle Testsequenzen.

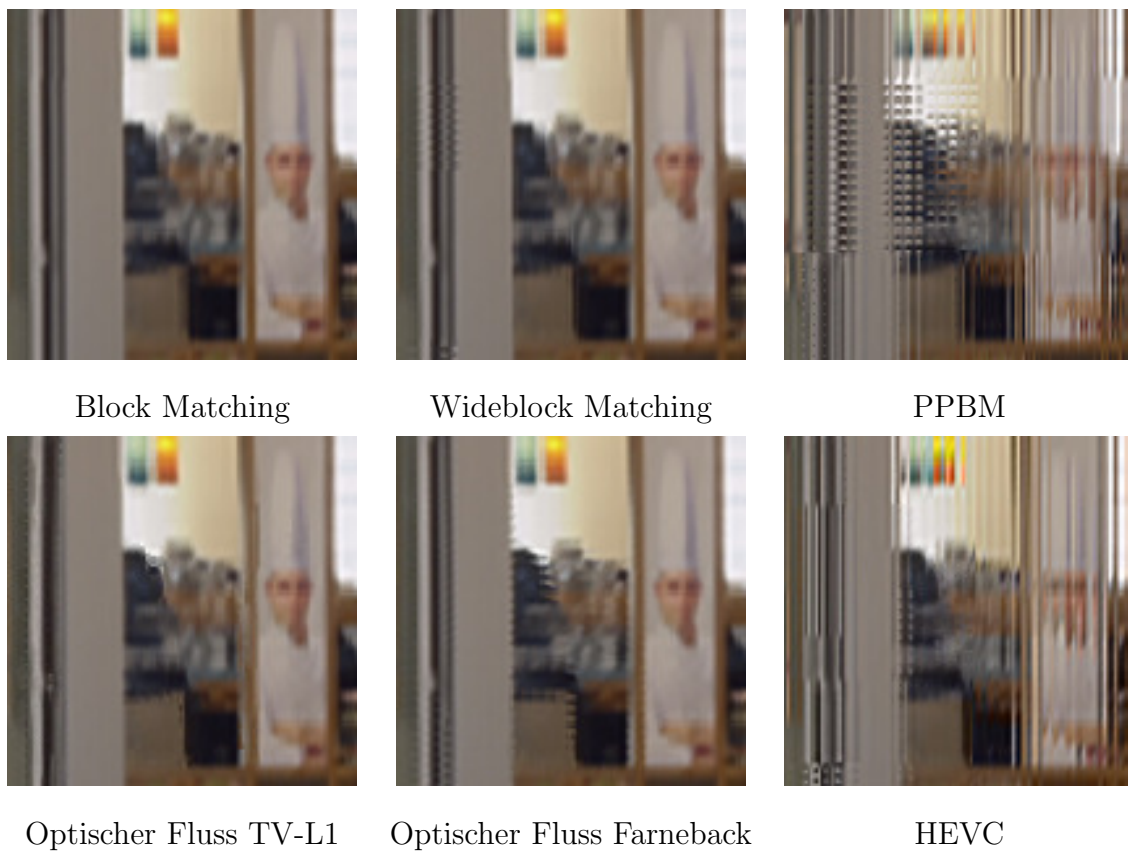


Tabelle 4.9: HYSR-Resultate für einen Ausschnitt der BQMall Sequenz mit 7 Referenzframes und dem jeweils besten Parameterset für die Bewegungsschätzung.

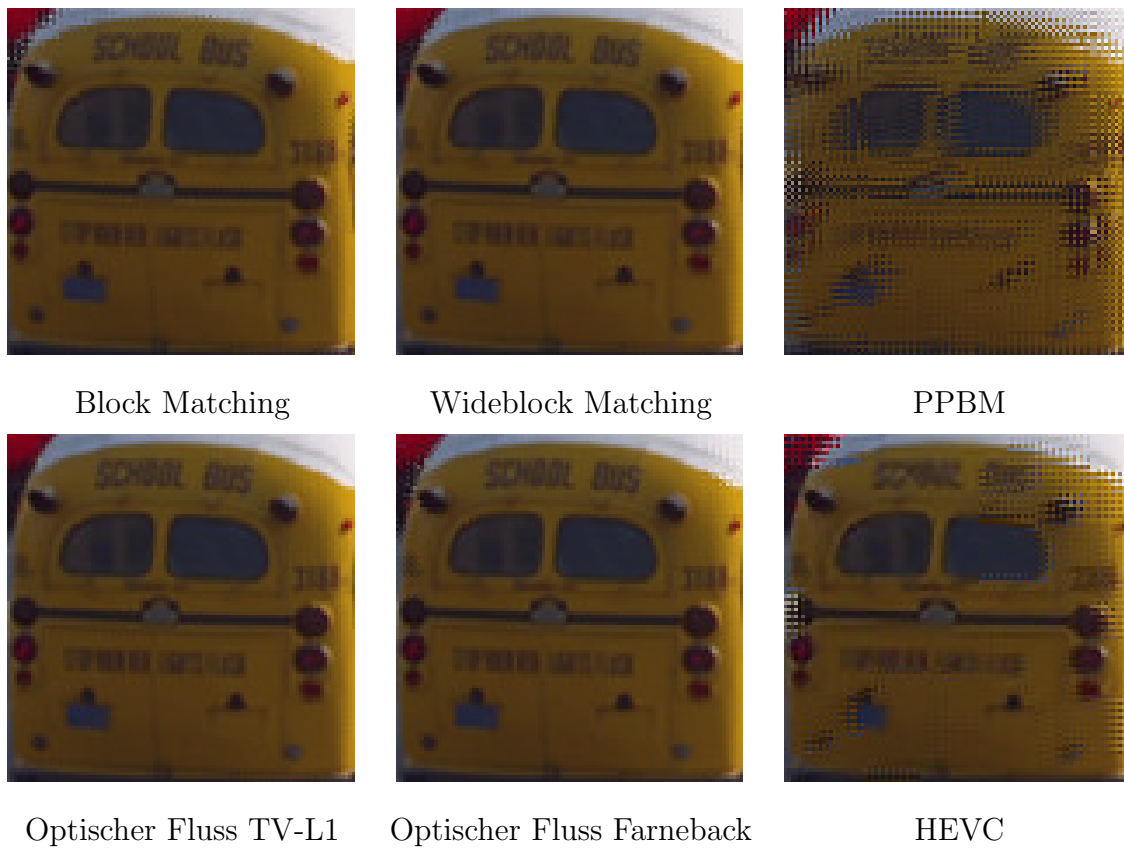


Tabelle 4.10: HYSR-Resultate für einen Ausschnitt der Traffic Sequenz mit 7 Referenzframes und dem jeweils besten Parameterset für die Bewegungsschätzung.

Kapitel 5

Ausblick

Die Untersuchung des Einflusses der Bewegungsschätzer Block Matching, Wideblock Matching, Parallel Predictive Block Matching, OpticalFlow TV-L1, OpticalFlow Farneback und HEVC auf ein HYSR-Verfahren hat gezeigt, dass sich unter Verwendung der Optischen Fluss Verfahren der höchste PSNR-Gewinn gegenüber einer bikubischen Interpolation erzielen lässt.

Es scheint lohnenswert in zukünftigen Arbeiten den Einfluss verschiedener Parameter-einstellungen in den Optischen Fluss Verfahren auf den PSNR-Gewinn zu untersuchen. Außerdem ließe sich der PSNR-Gewinn vermutlich auch durch eine Kombination der beiden Optischen Fluss Verfahren, abhängig von der verwendeten Anzahl an Referenz-frames, weiter steigern. (vgl. Kapitel 4.4)

Abbildungsverzeichnis

2.1	Ablauf des MISR Verfahrens	4
2.2	Rekonstruktionsschritt	5
2.3	Hybride Super Resolution Framework	6
3.1	Fullsearch Block-Matching Algorithmus	9
3.2	Aufbau eines Wideblocks	11
3.3	Position der zeitlichen und örtlichen Prädiktoren in PPBM	12
3.4	Sternförmige Bewegungsschätzung	13
3.5	Sternförmige Bewegungsschätzung mit Pre-PPBM Frame	14
4.1	Simulationsergebnisse aller Bewegungsschätzer	23
4.2	Vergleich der Parametersets für den PPBM gemittelt über alle Testsequenzen.	24
4.3	Vergleich der Bewegungsschätzer mit positivem PSNR-Gewinn gegenüber bikubischer Interpolation. Gemittelt über alle Testsequenzen und Parametersets.	25
4.4	Vergleich der Parametersets für den Optischen Fluss TV-L1 mit dem Optischen Fluss Farneback, gemittelt über alle Testsequenzen.	25

Tabellenverzeichnis

4.1	Simulierte Parametersets für Block Matching	20
4.2	Simulierte Parametersets für Wideblock Matching	20
4.3	Simulierte Parametersets für Parallel Predictive Block Matching	21
4.4	Simulierte Parametersets für Optical Flow TV-L1	21
4.5	Konstante Parameter jeder Simulation für Optical Flow TV-L1	21
4.6	Simulierte Parametersets für Optical Flow Farneback	22
4.7	Konstante Parameter jeder Simulation für Optical Flow Farneback	22
4.8	Simulierte Parametersets für HEVC	22
4.9	HYSR-Resultate für einen Ausschnitt der BQMall Sequenz mit 7 Referenzframes und dem jeweils besten Parameterset für die Bewegungsschätzung.	26
4.10	HYSR-Resultate für einen Ausschnitt der Traffic Sequenz mit 7 Referenzframes und dem jeweils besten Parameterset für die Bewegungsschätzung.	27

Literaturverzeichnis

- [1] Sung Cheol Park, Min Kyu Park, Moon Gi Kang, Super-Resolution Image Reconstruction: A Technical Overview, IEEE Signal Processing Magazine, 2003
- [2] K. I. Kim und Y. Kwon, Single-Image Super Resolution Using Sparse Regression and Natural Image Prior, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010
- [3] Michel Bätz, Andrea Eichenseer, Jürgen Seiler, Markus Jonscher, and André Kaup, Hybrid Super-Resolution Combining Example-Based Single-Image And Interpolation-Based Multi-Image Reconstruction Approaches,
- [4] Sandra Vedler, Analysis of a Multi-Reference Frame Motion Estimation Approach for Detecting the True Motion in Compressed Video Data, 2016
- [5] C. Zach, T. Pock, and H. Bischof, A Duality Based Approach for Realtime TV-L1 Optical Flow,
- [6] Berthold K.P. Horn and Brian G. Rhunck, Determining Optical Flow, 1980
- [7] Gunnar Farneback, Two-Frame Motion Estimation Based on Polynomial Expansion
- [8] http://docs.opencv.org/2.4/modules/video/doc/motion_analysis_and_object_tracking.html
- [9] Javier Sanchez, Enric Meinhardt-Llopis , Gabriele Facciolo TV-L1 Optical Flow Estimation,