



Schnelle Verfahren zur
Objektregistrierung in der
Bildverarbeitung am Beispiel der
Gesichtsstabilisierung

Diplomarbeit
zur Erlangung des Diplomgrades
Diplom-Informatiker (FH)

Fachhochschule Köln
Institut für Informatik - im Studiengang Allgemeine Informatik

Jannis Bloemendal

Erstprüfer: Prof. Dr. Wolfgang Konen
Zweitprüfer: Prof. Dr.Ing. M.Sc. Thomas Zielke

Gummersbach, im Februar 2009

Inhaltsverzeichnis

1	Einleitung	5
1.1	Motivation	5
1.2	Zielsetzung	6
1.3	Stand der Technik	6
1.4	Gliederung	8
2	Methoden	9
2.1	Transformation	9
2.2	Resampling	11
2.3	Interpolation	12
2.4	Kouroggi	12
2.5	Optimierungsansätze	15
2.5.1	Bounding Box	16
2.5.2	Beleuchtung	16
2.5.3	Ausreißer	18
2.5.4	Transformation	19
2.6	Pseudo Motion - Logarithmic Search	19
2.6.1	Logarithmic Search	19
2.6.2	Normierte Kreuzkorrelation	21
2.6.3	Neuer Algorithmus	22
2.7	Vergleichsmaß	25
2.8	Superresolution	27
2.8.1	Average-Image	29
2.8.2	Normalized Averaging	30
3	Testaufnahmen	33
4	Ergebnisse	35
4.1	Vergleichsmaß	35
4.1.1	Differenz	35
4.1.2	Differenz der Kantenstrukturen	38
4.1.3	Template-Korrelation	38

4.2	Face Bounding Box	42
4.3	Beleuchtung	43
4.4	Pseudo Motion Ausreißer	44
4.5	Interaktion der Optimierungsansätze	45
4.6	Neuer Algorithmus	47
4.7	Laufzeitvergleich	50
4.8	Superresolution	51
5	Zusammenfassung	57
6	Ausblick	58
7	Persönliches Schlusswort	58
8	Anhang	62
A	FaceSnapRecorder	62
B	Bounding Box im Gif Kommentar	70
	B.1 GIF-Format	70
	B.2 Bounding Box Editor	71
C	Aufnahmen	74

Abbildungsverzeichnis

1	Unterschied zwischen bewegtem Objekt und statischer Szene	7
2	Schematisches Beispiel Image Mosaicing	7
3	Beispiel zur Gesichtsstabilisierung	8
4	Bilineare Interpolation	13
5	Initiale Schätzung der Transformation durch die Bounding Box	17
6	Grauwertanpassung durch rekursiven tiefpass Filter	18
7	Logarithmic Search	20
8	Grafische Darstellung des neuen Algorithmus	24
9	$mean_{C_L}$ - Einteilung der Bilder in Templates	26
10	Schematische Darstellung einer Aufnahme durch einen CCD-Sensor	28
11	Raster zweier Frames nach der Registrierung	28
12	Normalized Averaging	31
13	Grenzen des Verfahrens	34
14	Künstlicher Fehler	36
15	Gegenüberstellung guter und schlechter Match	36
16	Gegenüberstellung guter und schlechter Match	38
17	Durchschnittsfehler für $contoure_{err}$	39
18	Spreizung des $contoure_{err}$	39
19	Variation der Templateanzahl für $mean_{C_L}$	41
20	Spreizung des $mean_{C_L}$	41
21	Variation der Tempaltezahl des $mean_{C_L}$ in Abhängigkeit von dem künstlichen Fehlergewicht w	42
22	Einfluss der Größe der Bounding Box auf das Verfahren von Kouroggi	43
23	Auswirkung der Subtraktion der Tiefpass Filterung für verschiedene Glättungsparameter a	44
24	Verteilung der Verschiebungsvektorbeträge	45
25	Einfluss der Frameanzahl auf das Average Bild	51
26	Vergleich der Profilplots der Referenz und Average-Bilder	52
27	Averaging-Verfahren mit Testmuster	54

28	Pixelvergleich Referenzbild - $2\times$ Average Bild	55
29	Ähnlichkeitsvergleich durch das <i>Comparison Window</i>	55
30	Betriebsart FaceSnapRecorder	64
31	Einstellung der Aufnahmeparameter für die Gesichtserkennung	65
32	Helligkeitseinstellungen	65
33	Search-modus zum Betrachten und Verwalten von Aufnahmen	67
34	Vergleich verschiedener Gesichter einer Gruppe mit der Galerie	68
35	Boundig Box Editor	72
36	Testset n004	81
37	Testset n006	81
38	Testset n013	81
39	Testset b001	81
40	Testset b004	82
41	Testset b009	82
42	Testset r002	82
43	Testset r000	83
44	Testset r016	83
45	Testset sr019	83

Tabellenverzeichnis

1	Gegenüberstellung der Vergleichsmaße	37
2	Vergleich der Optimierungen	46
3	Vergleich der initialen Methoden für den neuen Algorithmus .	49
4	Laufzeitvergleich	50
5	Verbesserung der Gesichtswiedererkennung durch die Average Bilder	56
6	Schematische Darstellung des GIF-Formats	73
7	Auszug aus den technischen Daten der Kamera	74
8	Beschreibung der Testsets	80

1 Einleitung

Objektregistrierung bzw. Bildregistrierung ist ein Verfahren der digitalen Bildverarbeitung, mit dem Ziel unterschiedliche Bilder aus einer Sequenz so zu transformieren, dass sie maximal ähnlich sind bzw. wenig Differenz aufweisen. Ein Bild der Sequenz wird hierfür als Referenzbild bezeichnet, zu dem alle anderen Bilder, (die wir Objekt nennen wollen), registriert werden. Eine geeignete Transformation zwischen einem Objektbild der Sequenz und dem Referenzbild zu finden ist ein Optimierungsproblem, das abhängig von einem Fehlermaß für die Güte der Differenz der Bilder ist[1].

1.1 Motivation

Bilder sind heutzutage ein wichtiges Beweismittel für kriminaltechnische Recherchen und Ermittlungen. Sie werden z.B. durch Videoüberwachungssysteme in vielen Situationen des alltäglichen Lebens erfasst und bieten umfangreiche Informationen. Diese Informationen aus Bildern und Videos zu extrahieren ist ein weites Feld der Bildverarbeitung, welches in den Bereich der Mustererkennung fällt.

In dieser Arbeit handelt es sich bei dem Muster um das Gesicht eines Menschen. Das Gesicht bietet individuell biometrische Informationen, mit denen sich ein Mensch identifizieren und als solcher erkennen lässt. Aus diesem Grunde sind Ausweisdokumente meist mit einem Lichtbild versehen, das mit zunehmender Technisierung auch in digitaler Form auf einem Chip abgespeichert wird [2].

Jedoch birgt die Erkennung, Aufzeichnung und Wiedererkennung von Gesichtern Hindernisse, die durch optische Störeinflüsse wie Beleuchtung und Begrenzungen durch die Digitalisierung gegeben sind. CCD-Sensoren sind durch Auflösung, Wertebereich sowie Abtastrate limitiert und rufen hierdurch Artefakte wie Blurring, Rauschen und Aliasing¹ hervor, die nicht dem

¹Wird auch als Moiré-Effekt bezeichnet und ist ein Muster, das in den Aufnahmen als Artefakt sichtbar ist. Es wird hervorgerufen durch eine zu geringe Auflösung einer Digitalkamera, die bei feinen Strukturen das Nyquist-Shannon-Theorem für die Abtastung des Signals nicht einhalten kann.

realen Bild entsprechen. Die Idee besteht darin, aus verschiedenen Aufnahmen eines Gesichts, die aus einer Sequenz stammen, mehr Informationen zu gewinnen, als ein Einzelbild der Sequenz bereit hält. Dieses wird als Super-resolution oder Noise Reduction bezeichnet [3][4].

1.2 Zielsetzung

Die Zielsetzung der Arbeit ist die Optimierung eines Objektregistrierungsverfahrens zur Gesichtsstabilisierung, um den visuellen Eindruck, sowie die Wiedererkennung einer Person zu verbessern. Der primäre Fokus ist zunächst auf die Optimierung des Registrierungsverfahrens gerichtet, dass für nachfolgenden Schritte und Verfahren eine essentielle Grundlage bildet. Eine Untersuchung hinsichtlich der Echtzeitfähigkeit der eingesetzten Verfahren und eine konkrete Implementierung könnten Gegenstand einer Folgearbeit sein.

1.3 Stand der Technik

Die Grundlage der Arbeit ist eine Matlab-Implementierung des echtzeitfähigen Algorithmus nach Kouroggi [5] und der FaceSnapRecorder des Unternehmens CrossMatch Technologies Inc., der Gesichter in einer Bild-Quelle erkennt und diese mit der Position und Größe als Kommentar im Gif-Format abspeichert. Der Algorithmus wird in seiner ursprünglichen Form für Image-Mosaicing eingesetzt. Ein Image-Mosaik entsteht aus einer Sequenz von Bildern einer relativ statischen Szene, die durch Bewegung bzw. Blickwinkeländerung der Kamera entstehen. Diese Bilder werden so transformiert, dass sie zusammengesetzt ein großes Bildmosaik ergeben (Abb. 2). Der Unterschied zu der Anwendung in dieser Arbeit besteht darin, dass hier eine statische Kamera benutzt wird, die Aufnahmen eines bewegten Objektes macht (Abb. 1). Die Aufnahmen der Objekte sollen so transformiert werden, dass sich die Objekte maximal decken (Abb. 3).

Es gibt nur wenige Image-Mosaicing-Algorithmen die automatisch arbeiten und zudem auch noch echtzeitfähig sind[5][6]. Die Grundlage der Arbeit bildet somit das Verfahren von Kouroggi [5].

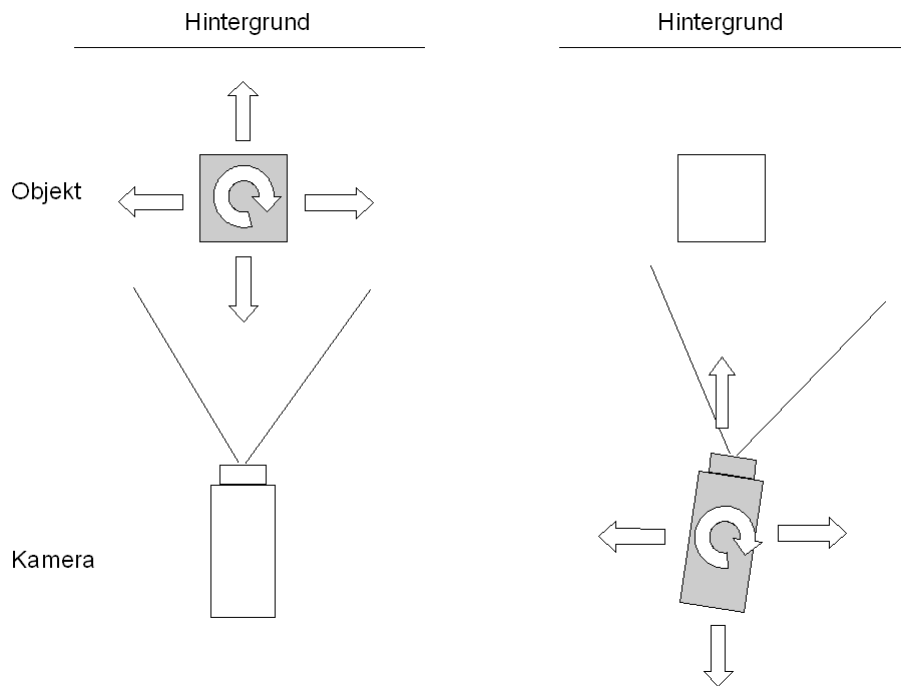


Abbildung 1: Unterschied zwischen bewegttem Objekt und statischer Szene



Abbildung 2: Schematisches Beispiel für Image Mosaicing: Ein Gesamtbild wird aus zwei einzel Bildern anhand des überlappenden Bereiches erzeugt.

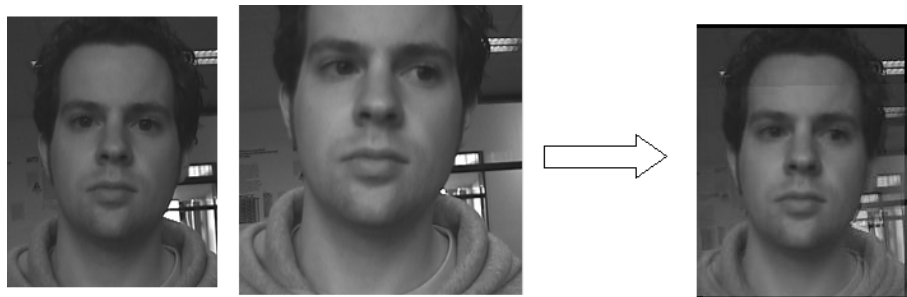


Abbildung 3: Beispiel zur Gesichtsstabilisierung: Links sind zwei Bilder aus einer Bildsequenz gezeigt. Das rechte Bild zeigt das Ergebnis der Gesichtsstabilisierung, bei der das mittlere Gesicht auf die Position des linken Gesichtes transformiert wurde und dann beide Bilder gemittelt wurden. Deutlich erkennt man die Überlagerung der Hintergrundstrukturen aus beiden Bildern, während die Gesichtsregion als eine Einheit wahrgenommen wird. (Die Registrierung wurde mit dem neuen Algorithmus durchgeführt [9])

1.4 Gliederung

In dem nachfolgenden Abschnitt 2 werden zunächst die Methoden erklärt, die zum Einsatz kommen. Zu dem Algorithmus von Kourogı (2.4) werden einige Optimierungsansätze (2.5) beschrieben. Im Abschnitt 2.6 wird hierauf ein neuer Algorithmus vorgestellt. Ein wesentliches Resultat dieser Arbeit stellt die Entwicklung von Vergleichsmaßen für die Objektregistrierung dar, die in Abschnitt 2.7 beschrieben wird. Um die vorgestellten Verfahren zu evaluieren, werden in Kapitel 3 Testaufnahmen beschrieben, die die Grundlage der Auswertungen in Kapitel (4) “Ergebnisse“ bilden. Mit dem Ergebnis der Auswertung wird dann in Kapitel 4.8 ein Superresolution-Bild aus einer Sequenz erzeugt.

2 Methoden

Nachfolgend werden die Methoden erläutert, die in den Experimenten zum Einsatz kommen.

2.1 Transformation

Transformationen sind geometrische Bildoperationen, die Pixel-Koordinaten eines Ausgangs-Bildes I neuen Positionen in dem Zielbild I'

$$I(x, y) \rightarrow I'(x', y') \quad \text{mit } x, y \in \mathbf{R} \quad (1)$$

zuordnen[7]. Um Koordinaten elegant durch Matrixmultiplikationen zu transformieren und projektive Abbildungen zu ermöglichen, werden homogene Koordinaten verwendet, die kartesische Koordinaten um eine Dimension erweitern:

$$\begin{pmatrix} \hat{x}' \\ \hat{y}' \\ h' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2)$$

Die homogenen Koordinaten werden durch Division von h' wieder in kartesische Koordinaten überführt:

$$\begin{aligned} x' &= \frac{\hat{x}'}{h'} \\ y' &= \frac{\hat{y}'}{h'} \end{aligned} \quad (3)$$

Die Koordinaten (x', y') sind reelle Zahlen, jedoch werden Bilder durch ein ganzzahliges Raster abgebildet, das durch die Auflösung definiert ist. Die Abbildung auf dieses Raster wird mit so genannten Interpolationsverfahren durchgeführt (siehe Abschnitt 2.3). Im Folgenden werden einige Transformationen beschrieben, die zum Einsatz kommen. Sie unterscheiden sich durch die Anzahl ihrer Freiheitsgrade a_{ij} .

Affine Transformation

Affine Transformationen fassen alle linearen Transformationen im 2-Dimensionalen Raum zusammen. Sie ermöglichen Translation, Rotation, Scherungen und Skalierung eines Bildes und bilden parallele Geraden wieder auf parallele Geraden ab.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (4)$$

Da $h' = 1$ ist, solange $a_{31} = 0$ und $a_{32} = 0$ sind, findet eine direkte Abbildung in kartesische Koordinaten statt.

Translation

Die Translation gehört zur Klasse der affinen Abbildungen mit zwei Freiheitsgraden, die das Ausgangsbild I um a_{13} in x und a_{23} in y Richtung verschiebt.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & a_{13} \\ 0 & 1 & a_{23} \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (5)$$

Rotations-Skalierungs-Translation

Eine affine Transformation ohne Scherung kann durch folgendes Gleichungssystem abgebildet werden:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ -a_{12} & a_{11} & a_{23} \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (6)$$

Projektive Transformation

Zusätzlich zu den sechs Koeffizienten der affinen Abbildung, werden für projektive Transformationen zwei weitere benötigt: a_{31} und a_{32} , wodurch die gesamte Transformations-Matrix in Gleichung (2) ausgeschöpft wird. Die

projektive Transformation vereint somit alle hier dargestellten Transformationen.

2.2 Resampling

Die Zuordnung der Koordinatenpaare, die durch die Transformation entstehen, kann auf zwei Arten erfolgen: Source-to-Target oder Target-to-Source. Das Ziel ist die reellen Koordinaten der Intensitätswerte auf ein neues diskretes Raster so anzuordnen, dass möglichst wenig Information verloren gehen ².

Source-to-Target

Bei der Source-to-Target Methode wird jeder Pixelkoordinate (u, v) im Ausgangsbild durch die Transformation T eine neue reelle Koordinate (u', v') im Zielbild zugeordnet.

$$(x', y') = T(u, v) \quad (7)$$

Für diese muss dann entschieden werden, welchen benachbarten Pixel-Koordinaten auf dem diskreten Raster der Intensitätswert zugeordnet werden soll. Das Problem hierbei ist, dass einzelne Koordinaten im neuen Bild, (zum Beispiel bei einer Vergrößerung), nie getroffen werden können und somit Lücken im Bild entstehen und bei einer Verkleinerung einzelne Koordinaten mehrfach adressiert werden können. Ein Lösungskonzept für die Vergrößerung, das ähnlich einer Interpolation ist, wird im Abschnitt 2.8.2 Normalized Averaging vorgestellt.

Target-to-Source

Target-to-Source Resampling ist das inverse Verfahren, bei dem zu jeder diskreten Pixelkoordinate (u', v') im Zielbild eine Ursprungskoordinate (x, y) berechnet wird.

$$(x, y) = T^{-1}(u', v') \quad (8)$$

Auch hier sind die durch die inverse Transformation entstandenen Koordinaten reellwertig. Der neue Intensitätswert wird durch ein Interpolationsver-

²vgl. [7] Seite 377-378

fahren (siehe Abschnitt 2.3) berechnet. Der Vorteil ist, dass hierbei keine Lücken entstehen, da für jeden Bildpunkt im Zielbild ein Ursprung berechnet wird.

2.3 Interpolation

Die Berechnung der Transformationen, sowie die Schätzung der Koeffizienten a_{ij} muss mit Subpixel-Genauigkeit durchgeführt werden, damit eine optisch gute Deckung der Bilder erreicht werden kann. Um diese zu erreichen wird biliniare Interpolation verwendet, wodurch Pixelkoordinaten nicht auf natürliche Zahlen bzw. das Raster der Auflösung begrenzt sind. Dies bildet die Grundlage der Superresolution Idee, die im Abschnitt 2.8 erläutert wird.

Gegeben ist die reelle Koordinate (u, v) , sowie die auf dem Raster benachbarten Intensitätswerte I_1, I_2, I_3 und I_4 . Gesucht ist der Intensitätswert I_7 , für den zunächst die Hilfswerte I_5 und I_6 berechnet werden (Abbildung 4).

$$\begin{aligned}
 I_5 &= I_1 + dx \cdot (I_2 - I_1) \\
 I_6 &= I_4 + dx \cdot (I_3 - I_4) \\
 I_7 &= I_5 + dy \cdot (I_6 - I_5) \\
 dx &= u - \lfloor u \rfloor \\
 dy &= v - \lfloor v \rfloor
 \end{aligned} \tag{9}$$

2.4 Kouroggi

Masakatsu Kouroggi et al. [5] haben ein Verfahren entwickelt, das ermöglicht, in Echtzeit Panorama-Mosaik-Bilder zu erzeugen. Hierfür wird eine zunächst grobe Schätzung der Verschiebungsvektoren für jedes Pixel des Bildes auf der Grundlage des optischen Flusses berechnet. Beruhend auf der Annahme, dass eine konstante Lichtquelle gleichbleibende Intensitätswerte in den Bildern erzeugt, werden für jedes Pixel des Bildes Raum-Zeit Gradienten des Grauwertes berechnet. Die Verschiebungsvektoren geben dann die Grundla-

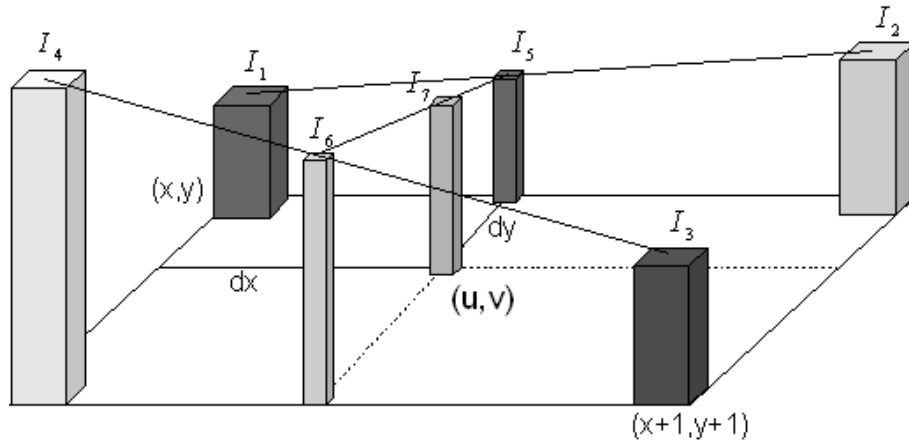


Abbildung 4: Bilineare Interpolation

ge um mittels der Methode der kleinsten Quadrate eine Annäherung an die reale Transformation zu berechnen.

Der Optische Fluss beschreibt die Positionsänderung jedes Intensitätswertes in einem Bild, die z.B. durch Bewegung des Betrachters oder der Kamera, (auch in einer ansonst statischen Szene), entsteht. Er wird beschrieben durch ein Verschiebungsvektorfeld

$$\begin{pmatrix} u(x, y) \\ v(x, y) \end{pmatrix} \quad (10)$$

, das für jeden Punkt (x, y) auf der Bildebene die Verschiebung für ein Zeitintervall Δt angibt. Unter der Annahme, dass sich die Lichtstärke im Zeitintervall Δt nicht ändert, gibt es zu jedem Intensitätswert $I(x, y)$ zum Zeitpunkt t eine Entsprechung $I(x + u, y + v)$ im Bild zum Zeitpunkt $t - \Delta t$.

$$I(x + u, y + v, t) - I(x, y, t - \Delta t) = 0 \quad (11)$$

Wenn man annimmt, dass die Intensität I eine stetige Funktion ist, so kann man für kleines u, v und Δt mittels der partiellen Ableitungen der Intensität

nach x, y und t eine Reihe entwickeln und erhält:

$$\begin{aligned}
I(x, y, t) + \frac{\partial I}{\partial x} \cdot u + \frac{\partial I}{\partial y} \cdot v - \left(I(x, y, t) - \frac{\partial I}{\partial t} \right) &= 0 \\
\frac{\partial I}{\partial x} \cdot u + \frac{\partial I}{\partial y} \cdot v + \frac{\partial I}{\partial t} &= 0
\end{aligned} \tag{12}$$

welches die Grundlage für das Verfahren nach Kouroggi und den dieser Arbeit zugrunde liegenden Algorithmus [6] bildet.

Kouroggi setzt in Gleichung (12) einfach $u = 0$ um v zu berechnen und vice versa. Zusätzlich wird eine so genannte compensated motion (u_c, v_c) berechnet, die eine grobe Schätzung für die gesuchte Position des Intensitätswertes darstellt und Diskontinuitäten im Grauwertverlauf kompensiert. Das Verschiebungsvektorfeld wird durch die so genannte Pseudo Motion beschrieben, die für jeden Pixel (x, y) durch

$$\begin{aligned}
\begin{pmatrix} u_p \\ v_p \end{pmatrix} &= \begin{pmatrix} -I_t^{(c)}/I_x \\ -I_t^{(c)}/I_y \end{pmatrix} + \begin{pmatrix} u_c \\ v_c \end{pmatrix} \\
I_t^{(c)} &= I(x + u_c, y + v_c, t) - I(x, y, t - 1)
\end{aligned} \tag{13}$$

berechnet wird. I_x und I_y sind hier die Gradienten in x- und y-Richtung und $I(t)$ und $I(t - 1)$ sind zwei aufeinander folgende Frames einer Sequenz[6]. Der Algorithmus startet mit $(u_c, v_c) = 0$ und berechnet die Pseudo Motion (u_p, v_p) iterativ in folgender Weise:

1. Berechnung der Pseudo Motion nach Gleichung (13) für jedes Pixel i
2. Nimm nur die Pixel für die weitere Berechnung, für die gilt: Es gibt einen optischen Fluss, so dass gilt $I_x \neq 0$ und $I_y \neq 0$, die gefundene Verschiebung $(x + u_p, y + v_p)$ liegt innerhalb der betrachteten Region und der gefundene neue Intensitätswert variiert nur um eine Konstante zum Beispiel $T = 5$:

$$|I(x + u_p, y + v_p, t) - I(x, y, t - 1)| < T \tag{14}$$

3. Berechne die Transformationskoeffizienten a_{ij} mittels der Methode der kleinsten Quadrate. Zum Beispiel für die Rotations-Skalierungs-Translation (2.1):

$$\begin{aligned} a_{11}x_i + a_{12}y_i + a_{13} &= u_{pi} \\ -a_{12}x_i + a_{11}y_i + a_{23} &= v_{pi} \end{aligned}$$

$$(A \cdot a - b)^2 = Min \quad (15)$$

$$\text{mit } A = \begin{pmatrix} x_1 & y_1 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ x_N & y_N & 1 & 0 \\ y_1 & -x_1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ y_N & -x_N & 0 & 1 \end{pmatrix}, \vec{a} = \begin{pmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{23} \end{pmatrix}, \vec{b} = \begin{pmatrix} u_{p1} \\ \vdots \\ u_{pN} \\ v_{p1} \\ \vdots \\ v_{pN} \end{pmatrix}$$

4. Berechne eine neue Schätzung für (u_c, v_c) mit Hilfe der Transformation \vec{a}

Die Iteration wird so lange wiederholt, bis eine feste Anzahl an Durchläufen erreicht ist, oder die Änderung im Verschiebungsvektorfeld (u_{pi}, v_{pi}) einen Grenzwert nicht mehr überschreitet.

2.5 Optimierungsansätze

Da das Verfahren von Kourogı für ein leicht anderes Problem konzipiert ist, als das Vorhaben dieser Arbeit, sind Anpassungen bzw. Optimierungen notwendig um es zu adaptieren.

2.5.1 Bounding Box

Die Bounding Box ist der Bereich im Bild, der von der Software FaceSnapRecorder als Gesicht erkannt wurde. Sie besteht aus einem für die Wiedererkennung der biometrischen Merkmale relevanten Ausschnitt des Gesichts und wird im Kommentar Block der Gif-Datei abgespeichert³. Die Bounding Box bietet somit die Information für eine ungefähre Lage und Größe der Region, die für die Registrierung relevant ist. Diese Information kann man nutzen um die initiale Schätzung der Transformation T für die Compensated Motion (u_c, v_c) zu berechnen bzw. das Objekt-Bild auf das Referenz-Bild zu skalieren und zu verschieben (Abb. 5). Gegeben sind die Höhe $s_i^{(x)}$ und Breite $s_i^{(y)}$, sowie das Zentrum (x_i, y_i) der Bounding Boxen, so kann mittels der Transformation T_i^{-1} die Bounding Box des Objektbildes i auf die des Referenz-Bildes abgebildet werden (Target-to-Source Resampling).

$$T_i^{-1} = \begin{pmatrix} s_i^{(x)}/s_1^{(x)} & 0 & x_1 - x_i \\ 0 & s_i^{(y)}/s_1^{(y)} & y_1 - y_i \\ 0 & 0 & 1 \end{pmatrix} \quad (16)$$

Wenn man die Registrierung nur anhand des Bereiches der Bounding Box durchführt, so wird der Hintergrund bzw. Teilregionen des Gesichts für die Registrierung außer Acht gelassen, welches einen positiven Effekt hervorbringen kann. Da die Bounding Box jedoch den Bereich für die Registrierung auch eingrenzt, kann man versuchen den Bereich zu variieren, indem man zusätzlich einen prozentualen Rand, bezogen auf die Abmessungen der Bounding Box hinzu- bzw. abzieht.

2.5.2 Beleuchtung

Die Annahme der konstanten Lichtquelle, auf der die Gleichung des optischen Flusses (12) beruht, ist jedoch in Realwelt-Sequenzen leider nicht immer erfüllt. Besonders durch bewegte Objekte, (wie hier das Gesicht eines Menschen), ist der Ausgangspunkt des gleichbleibenden Intensitätswertes in

³siehe Anhang B

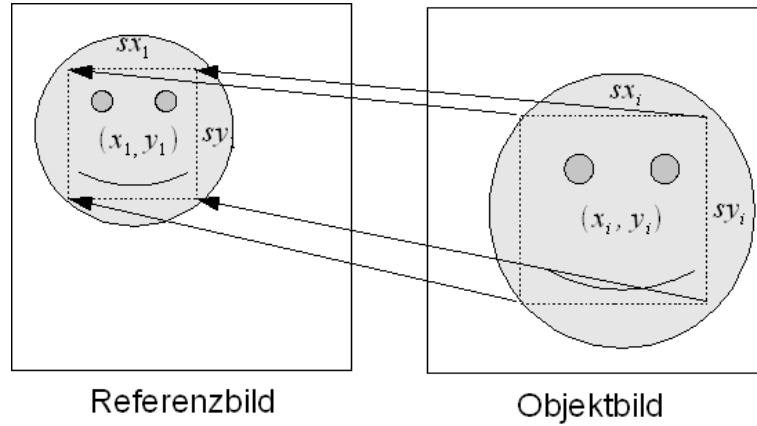


Abbildung 5: Initiale Schätzung der Transformation: Die Gesichtsregionen (Bounding Box) des Referenz- und Objektbildes geben durch ihre Lage und Größe eine grobe Schätzung für eine initiale Transformation des Objektbildes.

Regionen, die aufeinander abgebildet werden sollen, nicht immer gegeben. Menschen laufen in observierten Bereichen unter Lichtquellen hindurch und es gibt Lichteinfall aus mehreren Richtungen. Die Intensitätswerte variieren, die Gradienten führen nicht zu dem gewünschten Ziel und der Test in Gleichung (14) verhindert die Registrierung. Eine Möglichkeit die Intensitätswerte zu normalisieren ist von jedem Originalbild der Sequenz eine Tiefpassgefilterte-Version abzuziehen. Hierdurch werden die Kanten-Regionen hervorgehoben, die für einen Match der Bilder relevant sein können. Langsame Beleuchtungsgradienten werden ausgeglichen (Abb .6). Eine schnelle Implementierung ist mittels des rekursiven IIR-Filters[8] möglich, der zunächst für eine Dimension auf den Pixeln $k = 1, \dots, N - 1$ eine In-Place-Nach-Rechts-Rekursion

$$I_k = (I_k + aI_{k-1}) \frac{1}{1 + a}, \quad k = 1, \dots, N - 1 \quad (17)$$

und auf dem Ergebnis eine In-Place-Nach-Links-Rekursion

$$I_k = (I_k + aI_{k+1}) \frac{1}{1 + a}, \quad k = N - 2, \dots, 0 \quad (18)$$

mit dem Glättungsparameter a durchführt. Dieses wird für beide Dimensionen durchgeführt. Wird für a ein geringer Wert verwendet, so wird das

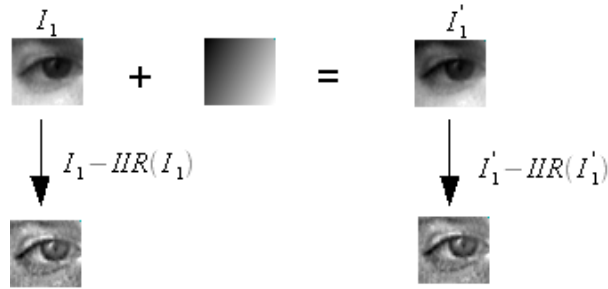


Abbildung 6: Grauwertanpassung durch rekursiven tiefpass Filter: Links ist die Addition eines linearen Grauwertverlaufs zu dem Ausgangsbild I_1 dargestellt. Von dem Ausgangsbild I_1 und dem Bild mit Beleuchtungsänderung I'_1 wird jeweils die gefilterte Version subtrahiert. Die Bilder I_1 und I'_1 werden hierdurch vergleichbar.

Ausgangsbild nur wenig geglättet, wohingegen $a \rightarrow \infty$ eine totale Mittelung bedeutet.

2.5.3 Ausreißer

Das Verfahren von Kouroggi versucht über den Test in Gleichung (14) Ausreißer auszusortieren. Jedoch werden Ausreißer, die eine falsche Zuordnung von Region des gleichen Grauwertes entsprechen nicht beachtet. Diese Falsch-Zuordnungen können durch große Gradienten entstehen, die die Iteration und somit die Compensated Motion in eine falsche Richtung lenkt. Um dieses zu vermeiden, sollte eine Betrachtung der Pseudo Motion Vektoren (u_p, v_p) durchgeführt werden, die den Test passieren, dessen geschätzte Verschiebung

$$\left| \begin{pmatrix} -I_t^{(c)}/I_x \\ -I_t^{(c)}/I_y \end{pmatrix} \right| < T_{maxmotion} \quad (19)$$

jedoch von der Gesamtheit abweicht. Durch eine Untersuchung des Betrages des Vektors kann eine Schranke gefunden werden, die eine maximale Länge $T_{maxmotion}$ für die Verschiebungsvektoren vorgibt und das Verfahren stabilisiert.

2.5.4 Transformation

Kouroggi verwendet affine Transformation in seinem Verfahren. Jedoch sollte auch untersucht werden, ob andere Transformationen für die Gesichtsstabilisierung bessere Ergebnisse erzielen kann. Zum Beispiel wäre die Rotations-Skalierungs-Translation eine Alternative mit weniger Freiheitsgraden oder die Projektive-Transformation eine Transformation mit mehr Freiheitsgraden.

2.6 Pseudo Motion - Logarithmic Search

Die bisherige Methode kann durch die Optimierungen schon entscheidend verbessert werden, jedoch ist sie nicht robust genug. Im Folgenden wird ein neues Verfahren nach Wolfgang Konen [9] vorgestellt, das auf normierter Kreuzkorrelation und Logarithmic Search basiert.

2.6.1 Logarithmic Search

Logarithmic Search ist eine Suchstrategie, die es ermöglicht, in $O(c \cdot \ln s)$ zu suchen[10]. Es ist ein greedy Verfahren mit überlappenden Subproblemen. Astrid Lundmark[11] schlägt hierfür Optimierungen hinsichtlich des Suchrasters und der Suchstrategie vor, die hier jedoch zunächst nicht berücksichtigt werden.

Der Algorithmus verfeinert mit zunehmender Iteration die Schrittweite und grenzt somit den Problemraum ein (Abb. 7). Wenn an einer Stelle das lokale Maximum gefunden ist, wird die Suche im näheren Umfeld der aktuellen Position fortgesetzt. Dieses wird so lange die Schrittweite $w > 1$ ist, bzw. bei Suche mit Subpixelgenauigkeit eine untere Grenze erreicht ist, wiederholt. Der Ablauf des Algorithmus ist wie folgt:

1. Beginne mit einer Startposition (u, v) und Schrittweite w im Objektbild
2. Suche innerhalb der Positionen $(u - w, v - w)$, $(u, v - w)$, $(u + w, v - w)$, $(u - w, v)$, (u, v) , $(u + w, v)$, $(u - w, v + w)$, $(u, v + w)$, $(u + w, v + w)$, diejenige mit maximaler Übereinstimmung. Ist die Position mit maximaler Übereinstimmung (u, v) , dann halbiere die Schrittweite $w = w/2$, sonst setze die Suchposition (u, v) auf die Position mit dem maximalem Wert

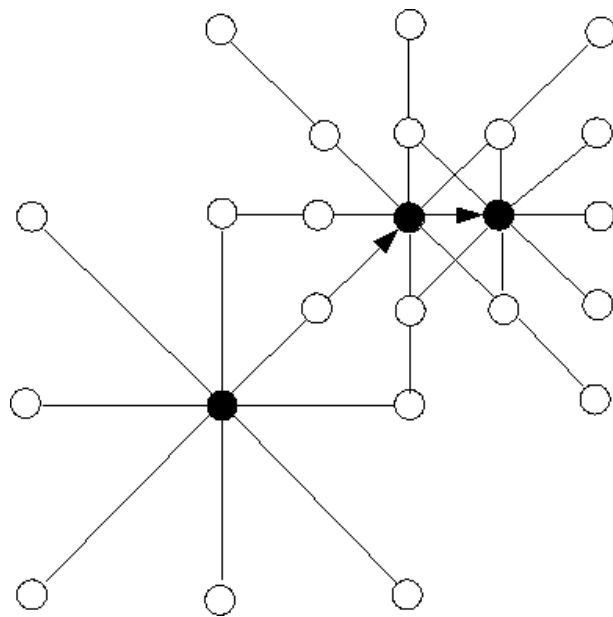


Abbildung 7: Schematische Darstellung des Logarithmic Search Algorithmus: Das Bild veranschaulicht die möglichen Schritte (Kreis) und die Wahl der besten Position (ausgefüllter Kreis). Der Algorithmus startet bei dem ausgefüllten Kreis unten links. Der Verlauf ist durch schwarze Pfeile gekennzeichnet. Im zweiten Schritt wird das lokale Suchfeld verfeinert.

3. Wenn $w > 1$ gehe zu 2.

Sei N die Anzahl an Positions-Auswertungen, i die Anzahl an Iterationen und SSF die Entscheidungsmöglichkeiten pro Iteration, so ist die Gesamt-Anzahl an Auswertungen:

$$N = i * SSF \quad (20)$$

Die Anzahl an Pixel s im Suchbereich ist:

$$\begin{aligned} s &= SSF^i \\ i &= \log_{SSF}(s) \end{aligned} \quad (21)$$

Durch Einsetzen von Gleichung (21) in (20) erhält man:

$$N = SSF \log_{SSF}(s) = \frac{SSF}{\ln(SSF)} \ln(s) = C_{SSF} \ln(s) \quad (22)$$

Die Komplexität ist proportional zum Logarithmus der Anzahl der Pixel und von dem Faktor C_{SSF} abhängig, der durch die Anzahl der Entscheidungsmöglichkeiten bzw. das Suchkreuz bestimmt wird[11].

2.6.2 Normierte Kreuzkorrelation

Normierte Kreuzkorrelation ist ein robustes Verfahren um in Intensitätsbildern die Distanz zwischen zwei Templates⁴ A und B im Referenz- und Objektbild zu bestimmen. Es leitet sich aus der Summe der quadratischen Abstände ab. Der Korrelationskoeffizient C_L ist durch die Normalisierung und den Bezug auf die lokalen Durchschnittswerte robust gegenüber globalen Intensitäts- und Kontraständerungen⁵. "Zusammengefasst ist unter realistischen Abbildungsverhältnissen der lokale Korrelationskoeffizient als zuverlässiges Maß für den intensitätsbasierten Bildvergleich zu empfehlen."⁶

$$C_L(A, B) = \frac{\sum_{m,n} (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_{m,n} (A_{mn} - \bar{A})^2)(\sum_{m,n} (B_{mn} - \bar{B})^2)}} \quad (23)$$

⁴Ausschnitt eines Bildes

⁵vgl. [7] S. 411-418

⁶[7] S.418

Er nimmt Werte im Bereich von -1 bis 1 an, wobei 0 keine Korrelation und 1 bzw. -1 eine hohe Korrelation bedeutet.

2.6.3 Neuer Algorithmus

Die Vorgehensweise besteht darin, für zwei gleich große Templates A und B des Referenz- und Objektbildes eine Entsprechung zu finden, sodass der Korrelationskoeffizient C_L maximal wird. Durch eine initiale Schätzung der Position eines Templates, lässt sich im Umfeld der Schätzung durch translatorische Verschiebungen ein optimaler Match anhand des maximalen Korrelationskoeffizienten ermitteln. Für die Suche des Matches wird das Logarithmic Search verfahren verwendet. Der Vorteil liegt darin, dass die Normierte Kreuzkorrelation Helligkeits- und Kontrastunterschiede sowie Rauschen kompensiert und so für Sequenzen mit Helligkeitsunterschieden und geringer Transformation gute Resultate erzielen kann. Sei durch $r_s = 2s + 1$ eine (ungerade) Templategröße definiert. Für eine minimale Anzahl an Landmarken⁷ $accept_{min}$ und die Begrenzung von Ausreißern durch u_{thresh} , ist die Vorgehensweise des Algorithmus:

1. Initiale Schätzung von (u_c, v_c)
2. Verteile $n \cdot n$ Landmarken L_l gleichmäßig auf dem Referenzbild
3. Für jede Landmarke $L_l = (x, y)$ finde im Umfeld von (u_c, v_c) mittels Log-Search die Position $(u_p, v_p) = (u_c + u_{adj}, v_c + v_{adj})$ im Objektbild I_i für die gilt:

$$\begin{aligned}
 C_L(A, B) &= \max, & \text{mit} \\
 A &= \{I_1(m, n) \mid x - s \leq m \leq x + s \wedge y - s \leq n \leq y + s\} \\
 B &= \{I_i(m, n) \mid x + u_p - s \leq m \leq x + u_p + s \wedge y + v_p - s \leq n \leq y + v_p + s\}
 \end{aligned} \tag{24}$$

⁷Position in dem Referenzbild, für die eine Entsprechung im Objektbild gesucht wird

und der Korrelations-Koeffizient ist größer als eine Schranke

$$C_L(A, B) \geq c_{min} \quad (25)$$

4. Wenn weniger als $accept_{min}\%$ der $n \cdot n$ Landmarken den c_{min} -Test bestehen, so nehme die $accept_{min}\%$ Landmarken mit dem größten $C_L(A, B)$.
5. Berechne mittels der Methode der Kleinsten Quadrate (Gleichung (15)) die Transformation a und hierdurch die Compensated Motion (u_c, v_c) neu.
6. Wenn der euklidische Abstand

$$\sqrt{(u_p - u_c)^2 + (v_p - v_c)^2} > u_{thresh} \quad (26)$$

zwischen der gefundenen Position (u_p, v_p) und der Schätzung (u_c, v_c) einer Landmarke größer ist als u_{thresh} , dann verwerfe sie.

7. Wenn weniger als $accept_{min}\%$ der $n \cdot n$ Landmarken den u_{thresh} -Test bestehen, so nehme die $accept_{min}$ Landmarken mit dem kleinsten euklidischen Abstand.
8. Berechne mittels der Methode der Kleinsten Quadrate (Gleichung (15)) die Transformation \vec{a}

Das Verfahren ist durch die Tests und die normalisierte Kreuzkorrelation relativ robust. Die Tests verhindern, richtig parametrisiert, Ausreißer, und der Kontrollparameter $accept_{min}$ verhindert, dass zu wenig Landmarken in die Berechnung der Transformation einfließen. Die initiale Schätzung von (u_c, v_c) kann durch eine begrenzte Anzahl an Iterationen durch das Verfahren von Kourogli oder durch die Position und Größe der Bounding Boxen, (siehe Abschnitt 2.5.1), durchgeführt werden. Desweiteren ist zu untersuchen, ob eine Transformation des Objektbildes auf das Referezbild durch die initiale Schätzung sinnvoll ist, damit etwaige Größenunterschiede in dem durch die Templates abgedeckten Bereiche vergleichbar sind.

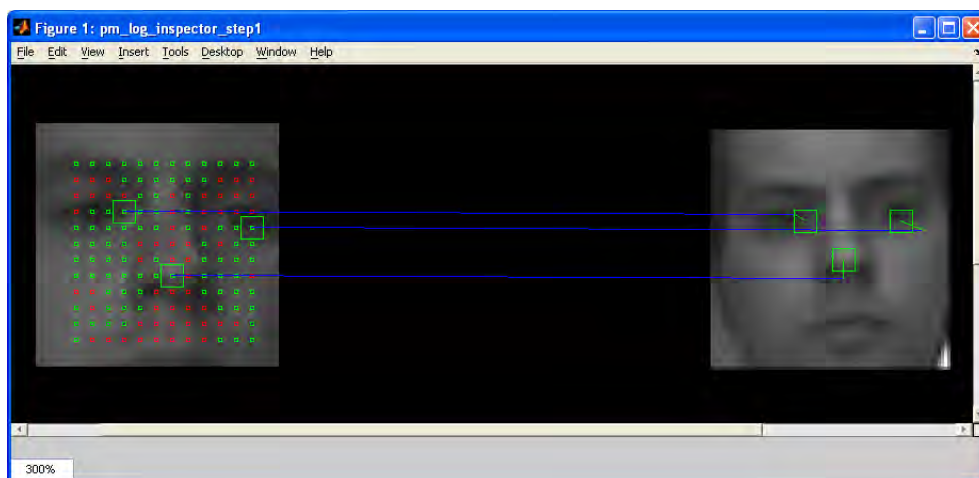


Abbildung 8: Grafische Darstellung des neuen Algorithmus: Durch das Tool *pm_log_inspector* kann die Arbeitsweise des neuen Algorithmus visualisiert werden. Die Landmarken sind durch grüne und rote Quadrate im Referenzbild dargestellt. Rote Quadrate bedeuten, dass für die Landmarke keine Entsprechung im Objektbild gefunden wurde, die die Tests passiert hat. Grüne Quadrate symbolisieren erfolgreiche Zuordnungen im Objektbild. Die blauen Linien zeigen die initialen Schätzungen für die Landmarken. Die grünen Linien zeigen von den initialen Schätzungen auf die gefundenen Positionen für die Landmarken.

In Abbildung 8 ist der Algorithmus durch das Tool *pm_log_inspector* grafisch dargestellt. Links ist das Referenz- und rechts das Objektbild angeordnet. Die Landmarken sind durch grüne und rote Quadrate abgebildet. Grün bedeutet, dass zu der Landmarke eine Entsprechung im Objektbild gefunden wurde, die die Tests bestanden hat. Die roten Landmarken werden nicht für die Berechnung der Transformation benutzt, da sie den Anforderungen nicht genügen. Als Beispiel wurden drei Landmarken ausgewählt, deren initiale Schätzung (u_c, v_c) durch einen blauen Vektor und die gefundene Position (u_p, v_p) durch einen grünen Vektor skizziert ist.

2.7 Vergleichsmaß

Um die Güte des Matches zwischen Referenz und registriertem Objektbild zu messen, benötigt man ein Fehlermaß, nachdem man das Registrierungsverfahren optimieren kann. Hier stellt sich die Frage, durch welches Verfahren der visuelle Eindruck wiedergegeben werden kann. Ein erster Ansatz besteht darin, die Differenz zwischen Referenzbild I_1 und registriertem Objektbild I_i^r innerhalb der Gesichts Bounding Box F_1 des Referenzbildes zu bilden und durch die Anzahl der Pixel $|F_1|$ zu teilen.

$$dist_{err} = \frac{\sum_{(x,y) \in F_1} (I_1(x,y) - I_i^r(x,y))}{|F_1|} \quad (27)$$

Globale Helligkeitsunterschiede werden hierbei nicht berücksichtigt und führen zu großen Fehlern, auch wenn der Match visuell gut ist. Eine andere Möglichkeit besteht auch hier darin, die Differenz zwischen zwei Bildern zu berechnen, die zuvor durch die Subtraktion ihrer Tiefpass-gefilterten-Version normiert wurden. Hierdurch werden Kanten und Strukturen in den Bildern betont und globale additive Helligkeitsunterschiede kompensiert (siehe Abschnitt 2.5.2).

$$contoure_{err} = \frac{\sum_{(x,y) \in F_1} ((I_1(x,y) - IIR(I_1(x,y))) - (I_i^r(x,y) - IIR(I_i^r(x,y))))}{|F_1|} \quad (28)$$

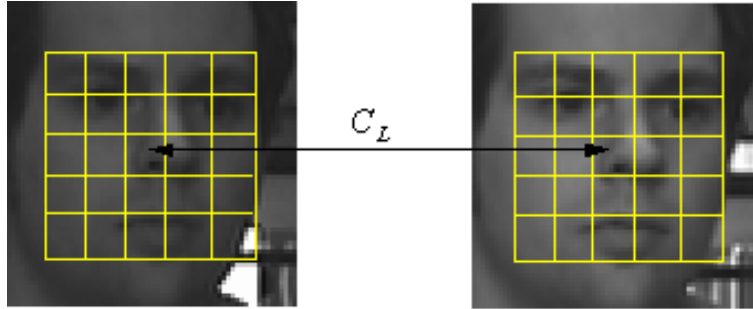


Abbildung 9: $mean_{C_L}$ Einteilung der Bilder in Templates: Die Bounding Box des Referenzbildes wird in gleich große Bereiche eingeteilt. Die Einteilung wird für Referenz- und registriertes Objektbild verwendet um die normierten Korrelationskoeffizienten für die einzelnen Ausschnitte (Templates) zu berechnen und über diese zu mitteln.

Eine weitere Möglichkeit besteht darin, den Match der Bilder durch den Normierten Korrelations Koeffizienten C_L (siehe Abschnitt 2.6.2) zu bewerten.

$$\begin{aligned}
 C_L(A, B) &= \frac{\sum_{m,n}(A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_{m,n}(A_{mn} - \bar{A})^2)(\sum_{m,n}(B_{mn} - \bar{B})^2)}}, & \text{mit} \\
 A &= \{I_1(x, y) \mid (x, y) \in F_1\} \\
 B &= \{I'_i(x, y) \mid (x, y) \in F_1\}
 \end{aligned} \tag{29}$$

Das Problem des Korrelationskoeffizienten ist, dass er die Energie des gesamten Bildes berücksichtigt. Jedoch können lokale Eigenschaften von einzelnen Ausschnitten durch eine Aufteilung der Bilder in kleinere Templates verglichen werden. Die Bounding Box wird hierfür in $n \times n$ gleich große Templates aufgeteilt und für alle n^2 zueinander gehörenden Templatepaare A_{ij}, B_{ij} der Korrelationskoeffizienten berechnet und gemittelt (Abb. 9).

$$\begin{aligned}
 mean_{C_L} &= \frac{\sum_{i=1}^n \sum_{j=1}^n C_L(A_{ij}, B_{ij})}{n^2}, & \text{mit} \\
 A_{ij} &\subset A & \text{und} \\
 B_{ij} &\subset B
 \end{aligned} \tag{30}$$

2.8 Superresolution

Die Übersetzung des englische Begriffs *“resolution“* ist Auflösung, die in der digitalen Bildverarbeitung die Fläche eines Bildes und somit die Anzahl der Pixel beschreibt. Je höher die Auflösung, desto höher die Dichte an Pixeln, die den Bildinhalt wiedergeben. Superresolution könnte bedeuten, dass man die Auflösung eines Bildes erhöht, um mehr Informationen zu bekommen. David Capel definiert in seiner Dissertation [3] den Begriff Superresolution weitreichender, indem er sinngemäß schreibt:

Ein super-resolution Bild gibt in der Wahrnehmung mehr Detailinformationen wieder, als ein low-resolution Bild. Dieses beinhaltet die Wiederherstellung von hochfrequenten Inhalten, für welche eine Erhöhung der Pixel-Dichte nötig sein kann⁸.

CCD-Sensoren bestehen aus einer fest angeordneten zweidimensionalen Matrix aus lichtempfindlichen Elementen, die Pixel ⁹ genannt werden[12]. In der Kamera wird das auf dem CCD-Sensor einfallende Licht durch Abtastung in digitale Werte quantisiert, die das Bild wiedergeben. Der Prozess unterliegt physikalischen Grenzen, die Rauschen, Aliasing oder auch Blurring hervorrufen können.

Wenn man nun davon ausgeht, dass das reale Bild der Szene unendlich viel Detail bereit hält, so wird dieses durch die Kamera in low-resolution Bilder verarbeitet (Abb. 10). Das Ziel ist nun, ein Superresolution Bild zu erzeugen, das mehr Informationen des originalen Bildes wiedergibt als jedes Einzelbild einer Sequenz.

Da das Objekt durch Bewegung immer wieder durch andere Elemente des CCD-Sensors abgebildet wird, kann man durch eine Subpixel-genaue Bestimmung der Transformation erreichen, dass einzelne Bildpunkte nicht mehr auf das durch den CCD-Sensor gegebene Raster fallen und es somit möglich wäre durch die Lage, sowie durch die zueinander gemappten Intensitätswerte, mehr Information zu extrahieren (Abb. 11).

⁸vgl. [3] S. 83

⁹Picture Element

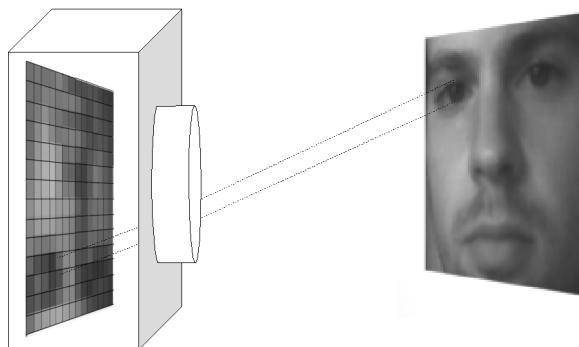


Abbildung 10: Schematische Darstellung einer Aufnahme durch einen CCD-Sensor: Die lichtempfindlichen Elemente des CCD-Sensors quantisieren das einfallende Licht in digitale Werte. Das Raster des Sensors liefert die reale Auflösung des aufgenommenen Bildes.

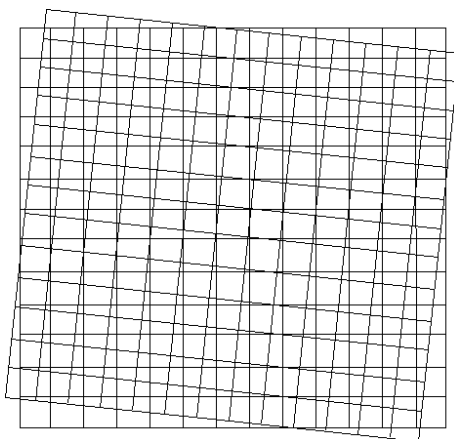


Abbildung 11: Raster zweier Frames nach der Registrierung: Einzelne Pixel des registrierten Objektbildes fallen nicht auf das Raster des Referenzbildes.

Eine ausführliche Diskussion verschiedener Verfahren soll an dieser Stelle nicht durchgeführt werden, da dieses nicht das Hauptziel der Arbeit ist. Jedoch werden im Folgenden zwei Verfahren vorgestellt, die einen Einstieg in das Thema der Superresolution geben.

2.8.1 Average-Image

Ein erster Ansatz in Richtung eines Superresolution Bildes ist das Durchschnittsbild I_{avg} aller registrierten Objekt-Bilder und dem Referenz-Bild zu erzeugen, welches eine Annäherung an ein geglättetes super-resolution Bild gibt. Es ist ein sehr robustes Verfahren um additives Rauschen der low-resolution Bildern zu unterdrücken ¹⁰.

Die Objekt-Bilder I_i werden mittels der gefundenen Transformation T_i auf das Referenz-Bild I_1 registriert. Die Transformation kann an dieser Stelle zuvor noch mit einer Skalierungs-Transformation T_{scale} verkettet werden, um die Auflösung des Average-Bildes I_{avg} zu erhöhen. Das Resampling wird hierfür Target-to-Source durchgeführt.

$$I_{avg} = \frac{\sum_{i=1}^n T_{scale} \circ T_i(I_i)}{n}, \text{ mit } T_{scale} = \begin{pmatrix} z & 0 & 0 \\ 0 & z & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (31)$$

Die Rauschunterdrückung kommt dadurch zustande, dass mit zunehmender Anzahl an registrierten Bildern I_i^r mit dem ungestörten Signal S_i , auch über den additiven Störanteil E_i des Signals der Mittelwert gebildet wird:

$$I_{avg} = \frac{\sum_{i=1}^n (S_i + E_i)}{n} = \bar{S} + \bar{E} \quad (32)$$

Unter der Annahme, dass der Störanteil E Normalverteilt ist mit $N(\mu, \sigma)$, würde lediglich der Mittelwert μ zu dem Mittelwert des Signals addiert werden.

¹⁰vgl. S. 97 ff. [3]

2.8.2 Normalized Averaging

Ein anderes Verfahren, das auf Source-to-Target Resampling beruht ein Superresolution Bild zu erzeugen, ist Normalized Averaging. Die Idee besteht darin, anstatt die Interpolation im Source-Bild durchzuführen, alle Pixel des Source-Bildes mittels Gewichtungen auf ein Superresolution-Raster zu transformieren. Die Lücken, die im Source-Bild durch das Source-to-Target Resampling entstehen, werden durch die Anzahl der Objekt-Bilder und eine Filter-Operation kompensiert. Tuan Q. Pham und Lucas J. van Vliet [13] haben ein Verfahren entwickelt, das durch Adaptive Filter auf der Grundlage des Normalized Averagings Superresolution Bilder erzeugt. Hier wird zunächst nur das Normalized Averaging mit einem Gauß-Filter von $\sigma = 1$ betrachtet, da es dem Averaging (Abschnitt 2.8.1) ähnlich ist.

Jedes Pixel der Objekt-Bilder wird durch Transformation auf eine reelle Superresolution Koordinate abgebildet und mittels gewichtetem Intensitätswert auf die vier nächsten Nachbarn des diskreten Rasters akkumuliert. Die Gewichte werden ähnlich eines Interpolations-Verfahrens gebildet und beziehen sich auf den Abstand der reellen Koordinate (x', y') zu dem diskreten Nachbarn des Rasters (Abb. 12).

$$1 = c_1 + c_2 + c_3 + c_4 \quad (33)$$

Es werden zwei SR-Raster SC_{accum} , C_{accum} verwendet, eines um die gewichteten Intensitätswerte $c \cdot I_i(x, y)$ aufzusummieren und eines für die Summation der Gewichtungen c . Durch eine Filteroperation ($*$) mit dem Faltungskern a wird der Einfluss der akkumulierten Intensitätswerte und Gewichte zueinander bestimmt. Anschließend wird über jedes Pixel der SR-Raster der Quotient gebildet, um das Bild I_{navg} zu berechnen.

$$\begin{aligned} SC'_{accum} &= SC_{accum} * a \\ C'_{accum} &= C_{accum} * a \\ I_{navg}(x, y) &= \frac{SC'_{accum}(x, y)}{C'_{accum}(x, y)} \end{aligned} \quad (34)$$

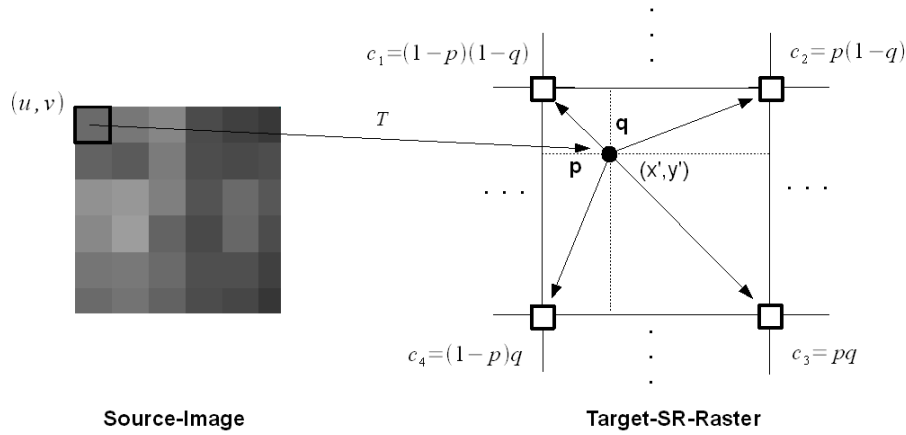


Abbildung 12: Normalized Averaging: Die Intensitätswerte werden durch Transformation auf reele Koordinaten des Zielbildes transformiert. Die Intensitätswerte der benachbarten Pixel des diskreten Ziel-Rasters wird durch die “Nähe“ der reelen Koordinate bestimmt.

Normalized Averaging ist ein sehr schnelles Verfahren und hat dadurch, dass nicht über die Anzahl der Pixel des Zielbildes iteriert werden muss, einen Geschwindigkeitsvorteil, abhängig von dem gewählten Vergrößerungsfaktor. Jedoch können, wenn nur wenige Frames zur Verfügung stehen, Lücken im Superresolution Bild auftauchen. Hier hat die Methode des Target-Source-Resamplings einen klaren Vorteil. Wenn man jedoch die Laufzeit für viele Frames betrachtet, so ergibt sich für die Anzahl der Operationen O_{avg} des Averaging Verfahrens bei n Bildern, der Anzahl s an Pixeln im Original-Bild und dem Skalierungsfaktor z :

$$O_{avg} = n \cdot s \cdot z^2 + s \cdot z^2 \quad (35)$$

Für die Anzahl der Operationen O_{navg} des normalized Averaging kommt noch eine Konstante f für die Faltung hinzu:

$$O_{navg} = n \cdot s + 2 \cdot f \cdot s \cdot z^2 + s \cdot z^2 \quad (36)$$

Bildet man den Grenzwert des Quotienten für $n \rightarrow \infty$, so erhält man:

$$\lim_{n \rightarrow \infty} \frac{O_{avg}}{O_{navg}} = \lim_{n \rightarrow \infty} \frac{z^2 \cdot s + \frac{s \cdot z^2}{n}}{s + \frac{2 \cdot f \cdot s \cdot z^2}{n} + \frac{s \cdot z^2}{n}} = z^2 \quad (37)$$

Das normalized Averaging ist also für eine große Anzahl an Frames um den Faktor z^2 schneller. Bei einer zweifachen Vergrößerung zum Beispiel um den Faktor 4.

3 Testaufnahmen

Um geeignete Testaufnahmen mit dem FaceSnapRecorder durchzuführen, ist zunächst das Ziel der Arbeit zu diskutieren, um somit gezielt Situationen zu erzeugen und diese zu evaluieren.

Das Ziel ist mehrere Bilder einer Sequenz auf ein Referenzbild zu registrieren. Die Bilder sollen hierfür relativ ähnlich sein. Das bedeutet, dass eine perspektivische Bewegung des Kopfes nur bis zu einem gewissen Grad abgebildet werden kann, da der Kopf bzw. das Gesicht nicht durch ein Verfahren erfasst wird, welches ein dreidimensionales Modell des Gesichtes berücksichtigt. Die primären Transformationen entstehen durch leichte Bewegung des Kopfes:

- Die Person bewegt sich auf die Kamera zu, oder von ihr weg (Skalierung)
- Leicht translatorische Bewegungen, die zum Beispiel durch das Laufen einer Person entstehen können
- Leichte Neigung des Kopfes nach links oder rechts (Rotation)
- Leichte Bewegung des Kopfes nach links oder rechts bzw. oben und unten (perspektivisch)

Bis zu einem gewissen Grad können perspektivische Bewegungen durch eine Transformationen der 2D-Bilder abgebildet werden. Jedoch, (wie auch in Abbildung 13 zu sehen), werden durch eine projektive Transformation zwar die Relationen richtig abgebildet, jedoch ist es hierdurch nicht möglich die wahre Perspektive der Kamera zu ändern. Obwohl das transformierte Bild einer Neigung des Kopfes nach unten entspricht, sind weiterhin die Nasenlöcher sichtbar. Dies veranschaulicht die Grenze des Verfahrens.

Die Affine Transformation ermöglicht neben der Skalierung, Rotation und Translation die Scherung von Bildern. Da eine Scherung als solche in den Test-Aufnahmen nicht vorkommt und das Verfahren der beschriebenen Grenze unterliegt, wird für die Registrierung zunächst nur die Rotations-Skalierungs-Translation 2.1 verwendet. Sie bietet den Vorteil, das sie über

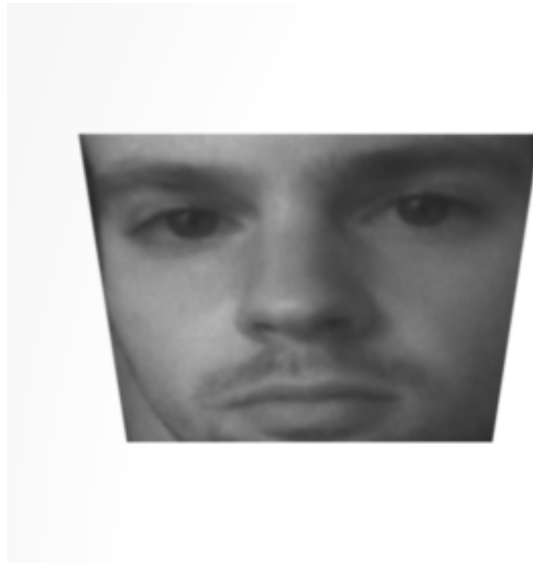


Abbildung 13: Grenzen des Verfahrens: Das Bild zeigt eine projektiv transformierte Aufnahme eines Gesichtes. Die Transformation entspricht einer Aufsicht auf das Gesicht, jedoch sind wie im Ausgangsbild immer noch die Nasenlöcher der Person sichtbar.

weniger Freiheitsgrade verfügt und die Bestimmung mittels der Methode der Kleinsten Quadrate somit robuster ist.

Weitere Szenarien sind, dass ein Gesicht durch die Bewegung unterschiedlichen Beleuchtungssituationen ausgesetzt ist. Diese können sowohl dadurch entstehen, dass eine Person unter einer Lichtquelle hindurch läuft, als auch durch eine sporadische seitliche Beleuchtung, zum Beispiel durch ein Hindernis zwischen Lichtquelle und Gesicht.

Ein anderer Umstand, den es zu verbessern gilt ist Rauschen in Bildern zu unterdrücken. Dieses kann zum Beispiel durch schlechte bzw. zu geringe Beleuchtung hervorgerufen werden.

Die Aufnahmen, die hier durchgeführt wurden, sind exemplarisch und sollen Möglichkeiten und Grenzen der Verfahren verdeutlichen. Im Anhang C befindet sich ein Auszug aus den Testsets und eine ausführliche Beschreibung. Die vollständigen Sets befinden sich auf der beigelegten DVD.

4 Ergebnisse

4.1 Vergleichsmaß

Um ein Vergleichsmaß (vgl. Abschnitt (2.7)) zu finden, welches der visuellen Wahrnehmung entspricht, werden visuell gute Matches den schlechten gegenübergestellt. Ein gutes Maß sollte einen Wert liefern, der zwischen guten und schlechten Matches unterscheidet und einen relativen Vergleich der Güte der Matches untereinander ermöglicht.

Aus den drei Testset-Kategorien ¹¹ wurden je 10 Sequenzen mit zwei Bildern ausgewählt, die die Grundlage zur Bestimmung eines Vergleichsmaßes bilden. Um den Einfluss durch die Auswahl der Matches zu minimieren und ein weites Spektrum an Fehlern abzudecken, wird in die visuell guten Registrierungen für die Bestimmung der Vergleichsmaße nachträglich ein gewichteter Fehler eingerechnet. Die Güte der Registrierung kann hierdurch künstlich gesteuert werden:

$$\begin{aligned} A'_i &= A_i \cdot \begin{pmatrix} \cos(\alpha) + s & -\sin(\alpha) & tx \\ \sin(\alpha) & \cos(\alpha) + s & ty \\ 0 & 0 & 1 \end{pmatrix}, \quad \text{mit} \\ \alpha &= \frac{\Pi}{16} \cdot r_k \cdot w && \text{(Rotationsfehler)} \\ s &= 0.2 \cdot r_k \cdot w && \text{(Skalierungsfehler)} \\ tx &= 9 \cdot r_k \cdot w && \text{(Translationsfehler)} \\ ty &= 9 \cdot r_k \cdot w && \end{aligned} \tag{38}$$

r_k ist eine standardnormalverteilte Zufallszahl und $w \in [0, 1]$ ein Gewicht um den Fehler zu beeinflussen (Abb. 14).

4.1.1 Differenz

Zunächst wird die Differenz zwischen zwei Bildern, (wie in Gleichung (27) beschrieben), als Vergleichsmaß untersucht. In Tabelle 1 sind gute ($w = 0$)

¹¹siehe Anhang C



Abbildung 14: Künstlicher Fehler: Durchschnittsbild des Referenz- und registrierten Objektbildes, in das nachträglich von links nach rechts folgende Fehler eingerechnet wurden: $w = 0; 0.3; 0.6; 0.9$



Abbildung 15: Gegenüberstellung eines guten und schlechten Matches: In die gute Registrierung ($w = 0$) des Test-Sets *n020* (links) wurde nachträglich ein gewichteter Fehler von $w = 0.6$ (rechts) hinzugefügt.

und schlechte Matches ($w = 0.6$) für verschiedene Vergleichsmaße gegenübergestellt. Die Differenz der Bilder grenzt für jedes Testset den guten vom schlechten Match ab. Es gibt keinen visuell schlechten Match, der einen besseren Fehlerwert erzielt, als ein visuell guter Match. Unter den Werten fallen jedoch einige auf, die in ihrer Abstufung nicht der visuellen Wahrnehmung entsprechen. Zum Beispiel ist die Differenz der Sets *n020* und *b007* zwischen gutem und schlechtem Match auffällig gering, der optische Eindruck spiegelt dieses jedoch nicht wieder (vgl. Abb. 15 und Abb. 16). Desweiteren ist auffällig, dass Beleuchtungsunterschiede wie im Testset *b004* zu großen Fehlerwerten führen.

Testset	$dist_{err}$		$contoure_{err}$		$mean_{C_L}$	
	good	bad	good	bad	good	bad
b002	0,86	32,11	8,43	42,87	0,85	0,21
b003	9,23	44,4	10,73	79,59	0,95	0,05
b004	34,92	59,56	31	55,01	0,66	0,23
b005	3,62	31,08	5,21	41,55	0,98	0,09
b006	0,43	11,19	7,3	23,18	0,93	0,06
b007	0	3,57	15,42	25,98	0,96	-0,02
b008	0,2	9,46	9,02	21,98	0,95	0,03
b009	33,95	39,6	15,69	28,12	0,95	0,33
b010	0,93	10,63	3,49	19,94	0,96	-0,01
b011	19,39	56,26	27,18	57,65	0,74	0,26
n002	2,39	21,74	7,26	34,38	0,94	0,25
n004	2,68	8,96	3,17	15,9	0,98	0,56
n006	2,83	17,73	5,28	29,11	0,97	0,34
n008	3,04	22,14	4,49	39,12	0,98	0,13
n010	1,24	16,22	1,54	18,39	0,99	0,06
n014	1,23	7,28	2,2	12,99	0,97	0,19
n016	1,58	11,06	2,73	18,41	0,98	0,17
n017	2,12	10,87	4,28	21,29	0,94	0,33
n019	1,81	11,08	3	21,01	0,98	0,49
n020	0,56	4,74	1,3	9,27	0,99	0,5
r002	4,81	9,23	9,78	20,73	0,71	0,27
r004	3,6	12,15	6,85	15,11	0,77	0,28
r005	3,55	16,8	8,35	20,76	0,77	0,17
r008	4,92	19,61	6,64	18,43	0,89	0,4
r009	6,43	7,98	13,77	17,29	0,73	0,65
r012	2,43	9,75	6,2	17,94	0,88	0,37
r013	3,04	13,47	11,71	28,45	0,84	0,23
r014	3,15	18,07	3,83	27,51	0,99	0,34
r015	1,53	7,23	3,82	10,7	0,9	0,26
r016	3,43	24,08	5,96	38,17	0,94	0

Tabelle 1: Gegenüberstellung der Vergleichsmaße: Auffällige Werte und Widersprüche sind hervorgehoben.



Abbildung 16: Gegenüberstellung eines guten und schlechten Matches: In die gute Registrierung des Test-Sets b007 (links) wurde nachträglich ein gewichteter Fehler von $w = 0.6$ (rechts) hinzugefügt.

4.1.2 Differenz der Kantenstrukturen

Um die Differenz der Kantenstrukturen als Fehlermaß, (wie in Gleichung (28) beschrieben), evaluieren zu können, muss zunächst ein geeigneter Glättungsparameter a (siehe Gleichung (17) und (18)) gefunden werden. Hierfür werden für verschiedene Parameterwerte a , jeweils der Fehler eines guten, sowie der Durchschnitt aus drei schlechten Matches ($w = 0,3; w = 0,6; w = 0,9$) für jedes Testset berechnet. Der Durchschnittsfehlerwert für eine Parametrisierung über das gesamte Vergleichsmaß-Test-Set gibt dann eine Abschätzung für verschiedene a -Werte.

Die Gegenüberstellung der Durchschnittsfehler (Abb. 17 und 18) zeigt, dass $a = 6$ den Fehler maximal spreizt.

Aus Tabelle 1 kann man entnehmen, dass die Abstufungen für einzelne Testsets gegenüber der Differenzmethode verbessert wurden. Der Fehlerwert liefert aber keinen absoluten Wert, um einen Match in der Güte mit anderen vergleichen zu können. Schlechte Matches erreichen zum Teil bessere Fehlerwerte, als visuell gute Matches anderer Testsets.

4.1.3 Template-Korrelation

Die paarweise Template-Korrelation (Gleichung (30)) ist durch die Anzahl der Templates parametrisierbar. Die Gegenüberstellung in Abbildung (19) verdeutlicht den Einfluss der Anzahl an Templates auf das Vergleichsmaß.

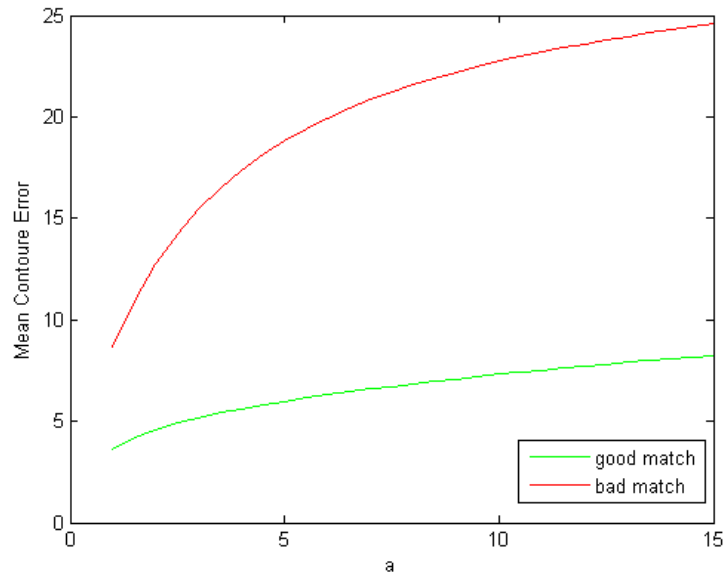


Abbildung 17: Durchschnittsfehler für $contoure_{err}$: Gegenüberstellung guter Match ($w = 0$) und Durchschnitt aus drei schlechten Matches ($w = 0, 3; w = 0, 6; w = 0, 9$) für verschiedene Glättungsparameter a .

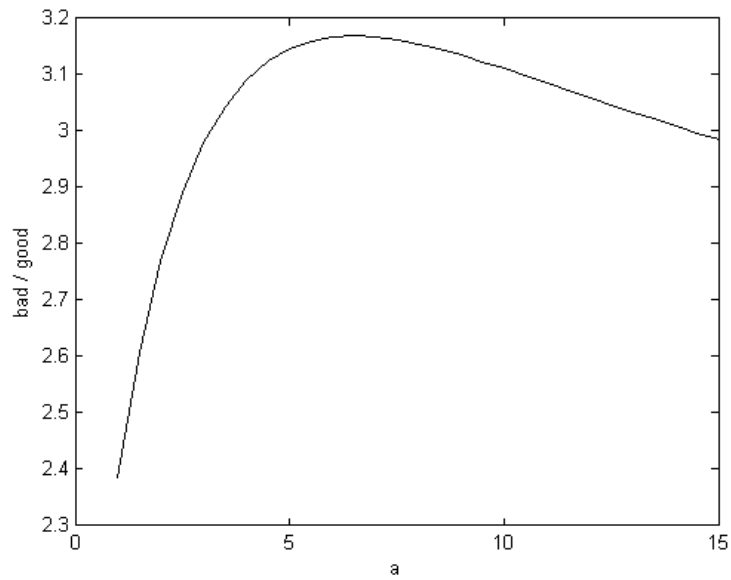


Abbildung 18: Spreizung des $contoure_{err}$ in Abhängigkeit von dem Glättungsparameter a .

Der Unterschied zwischen der Korrelation und den vorherigen Fehlermaßen ist, dass kleine Werte eine geringe Ähnlichkeit und große Werte eine hohe Ähnlichkeit bedeuten (vgl. Abschnitt 2.6.2). Die Abbildung 20 ist aus diesem Grund entgegengesetzt zu lesen: Der minimale Quotient liefert die maximale Spreizung für die Testsets. Das Maß spreizt sich bis zu einer Anzahl von 10 Templates pro Dimension ($10 \cdot 10$ insgesamt), danach nimmt die Spreizung nur noch geringfügig zu. Tests mit einer Anzahl von 10 Templates und mehr haben gezeigt, dass der visuelle Eindruck durch das Vergleichsmaß sehr gut wiedergegeben wird. Die Bilder der Testsets beinhalten aber naturgemäß leicht unterschiedliche Aufnahmen und der Unterschied der Aufnahmen zueinander wird, unabhängig von der Registrierung, auch durch das Fehlermaß abgebildet. Dieses fällt bei 10 Templates mehr ins Gewicht als zum Beispiel bei 5. Aus diesem Grund liefern 5×5 Templates zwar nicht die optimale Spreizung, aber der visuelle Eindruck für die Registrierung entspricht eher dem Fehlermaß. In Abbildung 21 ist das Verhalten für die Aufteilung der Bounding Box in unterschiedliche Anzahl an Templates pro Dimension abgebildet. Wird die Bounding Box nur durch ein Template verglichen (C_L), so wird bei einem künstlichen Fehler von $w = 1$, welcher visuell nicht mehr als Match bezeichnet werden kann, noch ein durchschnittlicher Korrelationskoeffizient von $mean_{C_L} = 0.28$ erreicht. Eine Anzahl von 10 Templates pro Dimension, bildet hingegen visuell gute Matches mit einem durchschnittlichen Korrelationskoeffizienten von $mean_{C_L} = 0,81$ ab, 5 Templates pro Dimension erreichen für gute Registrierungen einen $mean_{C_L} = 0.9$. Der Fehlerverlauf fällt mit zunehmenden Fehlergewicht stärker ab und ein Fehlergewicht von $w = 0.3$ wird durchschnittlich auf $mean_{C_L} = 0.47$ abgebildet (vgl. Abb. 14).

Aus der Tabelle 1 kann man entnehmen, dass die Abstufungen der Güte der Matches, im Vergleich zu den anderen Vergleichsmaßen, auch zwischen den Testsets gegeben ist. Für die weitere Betrachtung und Optimierung wird daher der durchschnittliche Template-Korrelations-Koeffizient mit einer Einteilung der Bounding Box in 5×5 Templates verwendet.

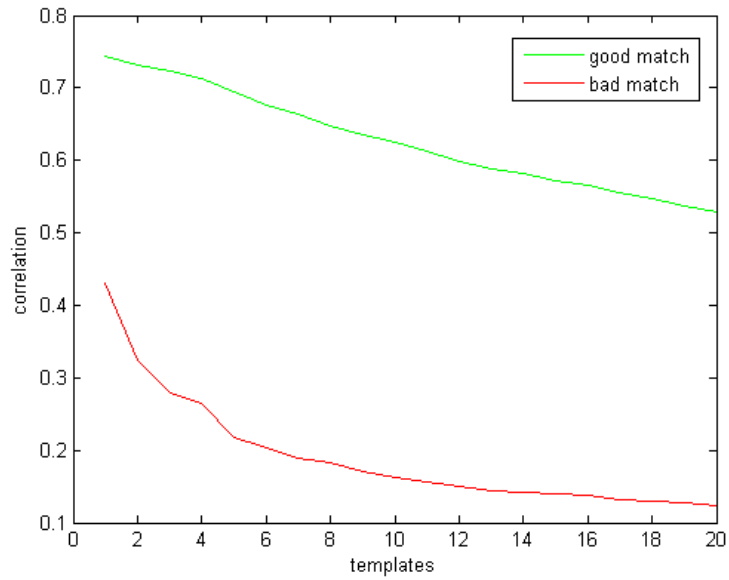


Abbildung 19: Variation der Templateanzahl: Durchschnittlicher $mean_{C_L}$ für gute Matches ($w = 0$) und dem Durchschnitt aus drei schlechten Matches ($w = 0, 2; w = 0, 4; w = 0, 6$).

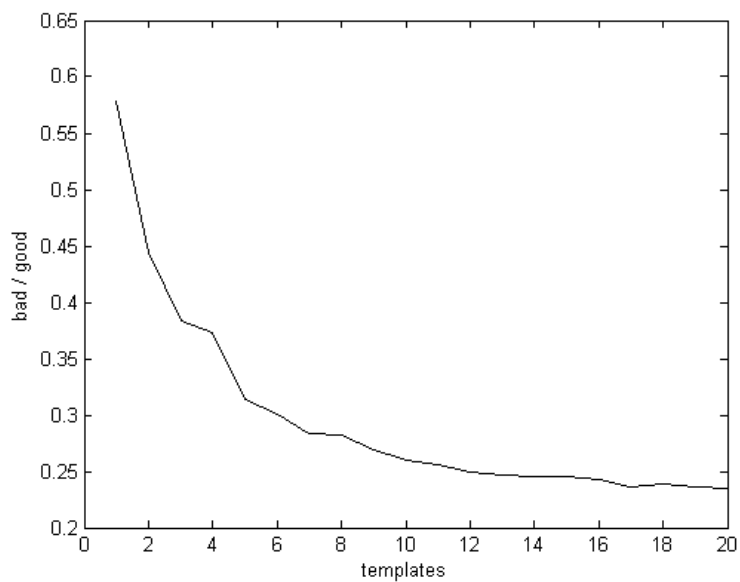


Abbildung 20: Spreizung des $mean_{C_L}$ in Abhängigkeit von der Anzahl an Templates.

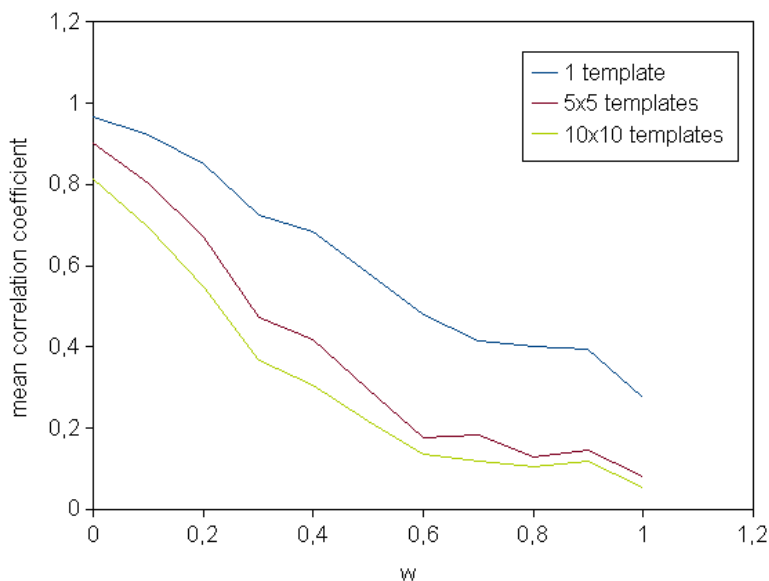


Abbildung 21: Variation der Tempaltezahl des $mean_{C_L}$ in Abhängigkeit von dem künstlichen Fehlergewicht w .

4.2 Face Bounding Box

Im Abschnitt 2.5.1 wurde beschrieben, wie die Bounding Box genutzt werden kann um die Registrierung zu optimieren. Der Einsatz der Bounding Box verbessert die Registrierung (vgl. Tabelle 2) im Durchschnitt für alle Testsets. Es fallen aber einige Problemfälle heraus, die durch den Einsatz der Bounding Box schlechter registriert werden.

Der relativ schlechte Durchschnittswert von $mean_{C_L} = 0,53$ über alle Testsets mit Einsatz der Bounding Box spiegelt sich auch in der Betrachtung der Resultate wieder. In den meisten Sequenzen sind nur die ersten zwei bis drei Objektbilder relativ gut gematcht. Sequenzen mit Intensitätsdifferenzen und hohem Rauschanteil stellen für den Algorithmus von Kouroggi ein Problem dar.

Für die weitere Untersuchung der Verfahren wurde aus den Testsequenzen eine Auswahl getroffen, so dass aus jeder Bildkategorie ¹² je fünf Bildpaare vorhanden sind, die mit der Verbesserung durch den Einsatz der Bounding

¹²siehe Anhang C

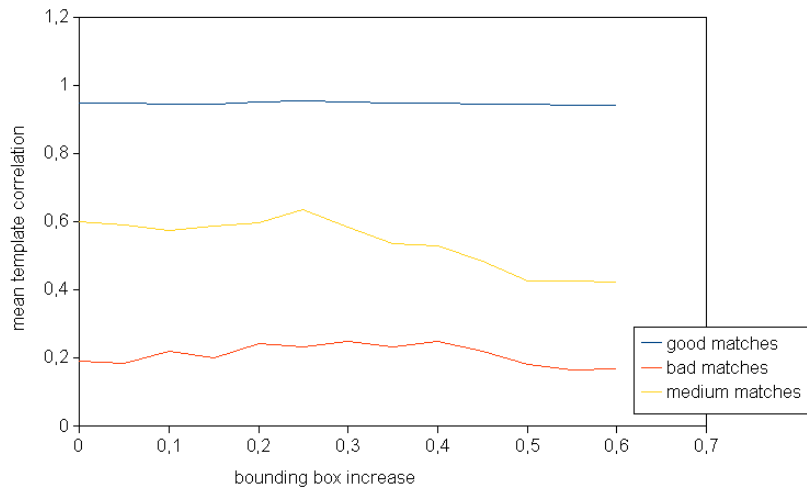


Abbildung 22: Einfluss der Größe der Bounding Box auf das Verfahren von Kouroggi.

Box den folgenden Kriterien bzw. Fehlerwerten entsprechen:

1. $0.8 < mean_{C_L} \leq 1$: guter Match
2. $0.4 < mean_{C_L} \leq 0.8$: mittel schlechter Match
3. $mean_{C_L} \leq 0.4$: schlechter Match

Die Einteilung soll eine differenzierte Betrachtung der Auswirkung der Methoden ermöglichen.

Die Vergrößerung der Bounding Box hat nur geringfügig Einfluss auf das Verfahren (Abb. 22). Eine Vergrößerung von 25% verbessert die Registrierung der mittel-schlechten Matches, wird die Region weiter vergrößert, führt dieses zu einer kontinuierlichen Verschlechterung der Registrierung der vorher schlechten und mittelschlechten Matches.

4.3 Beleuchtung

Die Differenz der Intensitätswerte, die durch unterschiedliche Beleuchtungssituationen in den Aufnahmen entstehen kann (vgl. Abschnitt 2.5.2), stellt für den Algorithmus von Kouroggi ein Problem dar. Die Subtraktion der gefilterten Bilder verbessert für $a \geq 2$ die vorher schlechten Matches, jedoch werden

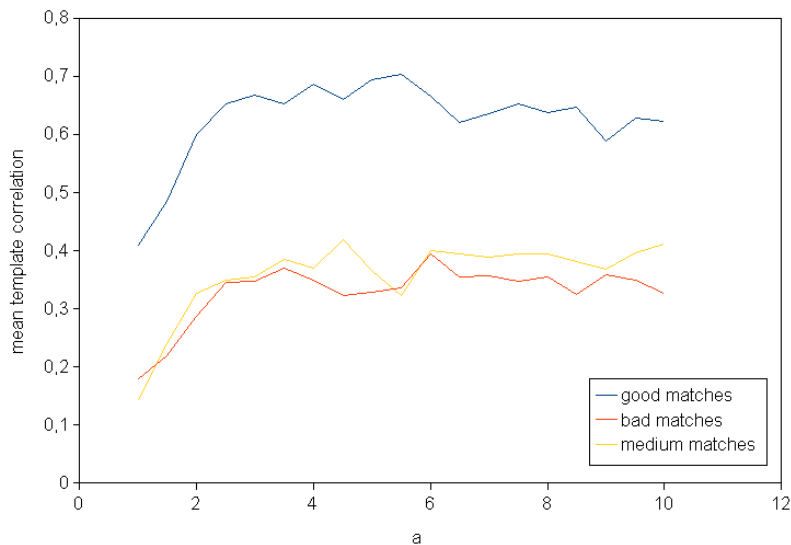


Abbildung 23: Auswirkung der Subtraktion der Tiefpass Filterung für verschiedene Glättungsparameter a auf das Verfahren von Kourogı.

hierdurch die vorher mittel-schlechten Matches, die zuvor einen durchschnittlichen $mean_{C_L} = 0.58$ erreicht haben deutlich verschlechtert, so dass der Einfluss auf die Registrierung insgesamt als negativ zu bewerten ist (vgl. Abb. 23 und Tabelle 2).

4.4 Pseudo Motion Ausreißer

Die Bestimmung eines Grenzwertes für den maximalen Betrag der Verschiebungsvektoren (vgl. Abschnitt 2.5.3) wird durch eine Stichprobe von 12 Testsets ermittelt. Diese besteht zu je einem Drittel aus den vorher als gut, mittel-schlecht und schlecht eingestuften Matches. In Abbildung 24 sind die Beträge der Vektoren für die gesamten Durchläufe der Iterationen über die Testsets in einem Boxplot dargestellt (das erste Quartiel liegt bei 0.90, der Median ist 2.41, das 3.Quartiel 5.65 und der Maximalwert 78,02). Anhand des oberen Whisker kann eine Grenze für Außerreißer abgelesen werden. Der Wert 15 liefert somit eine obere Schranke für einen maximalen Betrag der Verschiebungsvektoren. Die Auswertung in Tabelle 2 zeigt, dass durch den Einsatz der Begrenzung eine deutliche Verbesserung erzielt werden kann.

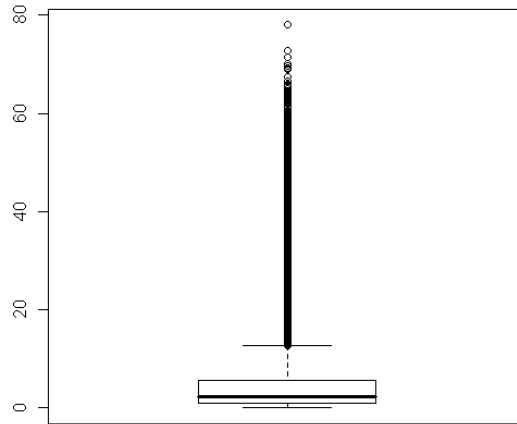


Abbildung 24: Verteilung der Verschiebungsvektorbeträge für eine Auswahl von je sechs Bildpaare aus den guten, mittel-schlechten und schlechten Matches

4.5 Interaktion der Optimierungsansätze

In der Tabelle 2 ist die Interaktion der Optimierungsansätze veranschaulicht. Hier wird ersichtlich, dass die zuvor als schlecht eingestufte Methode der Intensitätsanpassung durch den Tiefpassfilter in Kombination mit der Begrenzung der Ausreißer als bisher beste Methode hervorgeht. Sie erreicht für die gesamten Kategorien gute Resultate und bewirkt für einige Testsets eine Änderung der Einstufung in die nächst bessere Kategorie.

Method	prev. good		prev. medium		prev. bad		all	
	mean	std	mean	std	mean	std	mean	std
Kouroggi	0,84	0,21	0,36	0,28	0,23	0,22	0,48	0,36
Kouroggi, Bounding Box	0,95	0,03	0,58	0,06	0,05	0,1	0,53	0,38
Kouroggi, Bounding Box+0,25%	0,96	0,03	0,65	0,22	0,22	0,22	0,61	0,35
Kouroggi, Bounding Box, Maxmotion	0,98	0,03	0,75	0,09	0,47	0,27	0,72	0,26
Kouroggi, Bounding Box+0.25%, Maxmotion	0,97	0,03	0,76	0,16	0,51	0,23	0,74	0,24
Kouroggi, Bounding Box, IIR	0,96	0,31	0,36	0,38	0,31	0,33	0,45	0,38
Kouroggi, Bounding Box+0,25%, IIR	0,98	0,24	0,34	0,34	0,26	0,33	0,48	0,4
Kouroggi, Bounding Box, IIR, Maxmotion	0,98	0,05	0,79	0,14	0,75	0,19	0,82	0,16
Kouroggi, Bounding Box+0.25%, IIR, Maxmotion	0,98	0,03	0,83	0,12	0,75	0,25	0,84	0,18

Tabelle 2: Vergleich der Optimierungen: Mittelwert und Standardabweichung des $mean_{C_L}$ für das Testset.

4.6 Neuer Algorithmus

Der neue Algorithmus (Abschnitt 2.6) kann durch zahlreiche Möglichkeiten parametrisiert werden. Die Parameter des Algorithmus werden, wie von Wolfgang Konen [9] ermittelt, benutzt. Die Grundlage bzw. Startposition (u_c, v_c) der Suche wird durch ein anderes Verfahren gegeben, dessen Einfluss genauer untersucht wird. Für die initiale Schätzung der Suchpositionen bieten sich zwei Ansätze an:

1. Eine begrenzte Anzahl an Iterationen mit der Methode von Kouroggi und den zuvor ermittelten Optimierungen
2. Die Größe und Lage der Bounding Boxen (vgl. Abschnitt 2.5.1)

Die Anzahl der Iterationen hat Einfluss auf das gesamte Verfahren. Durch mehr Iterationen wird eine bessere Grundlage für den neuen Algorithmus geschaffen, jedoch wirkt sich dieses negativ auf die gesamte Laufzeit aus. Aus diesem Grunde werden zunächst fünf Iterationen verwendet.

Das Verfahren von Kouroggi wurde bisher mit der Rotations-Skalierungs-Translation verwendet. Für eine initiale Schätzung der Suchposition kann auch eine Translation verwendet werden. Da die Objektbilder in der Regel nur geringfügig rotiert und skaliert sind, kann eine reine Translation für die Bestimmung von (u_c, v_c) robuster sein. Etwaige Größenunterschiede der Bilder verschlechtern den Templatevergleich jedoch hierdurch. Die Start-Suchpositionen liegen nicht in der Region, in der eine Suche zum Ziel führen kann und der Vergleich von Templates gleicher Größe in verschiedenen großen Bildern führt zu nicht vergleichbaren Ausschnitten.

Das Verfahren, mit der die initiale Transformation geschätzt wird, kann zusätzlich benutzt werden um das Objektbild zuvor zu transformieren. Die initiale Transformation verkettet mit der Transformation, die hierauf durch den neuen Algorithmus ermittelt wird, liefert dann die gesuchte Transformation des Objektbildes.

In Tabelle 3 sind die Ergebnisse der verschiedenen Ansätze sowie das Verfahren von Kouroggi, mit der vorher ermittelten Optimierung für das Testset, aufgeführt. Die initiale Transformation liefert als Grundlage für den neuen

Algorithmus gute Resultate. Der Durchschnitt aller Kategorien kann durch das neue Verfahren in den Bereich der guten Matches angehoben werden. Die zuvor als mittel-schlecht und schlecht eingestuften Testsets erreichen in etwa gleich gute Korrelationen. Die Standardabweichung und der Mittelwert für die zuvor schlechten Matches wird sogar auf $mean = 0,86$ und $std = 0,1$ verbessert.

Method	prev. good		prev. medium		prev. bad		all	
	mean	std	mean	std	mean	std	mean	std
Kourogι, Bounding Box+0.25%, IIR, Maxmotion	0,95	0,03	0,83	0,12	0,75	0,25	0,84	0,18
Log-Search, Bounding Box+0.25%, Kourogι, IIR, Translation-Warp	0,96	0,03	0,86	0,09	0,84	0,16	0,89	0,12
Log-Search, Bounding Box	0,86	0,25	0,73	0,3	0,64	0,43	0,74	0,35
Log-Search, Bounding Box+0.25%	0,89	0,23	0,73	0,31	0,63	0,37	0,75	0,33
Log-Search, Bounding Box+0.25%, Kourogι, IIR, Trans-Scal-Rot-Warp, Initial Transformation	0,96	0,03	0,88	0,08	0,86	0,11	0,9	0,09
Log-Search, Bounding Box+0.25%, Initial Transformation	0,97	0,03	0,87	0,09	0,86	0,11	0,9	0,1
Log-Search, Bounding Box, Initial Transformation	0,96	0,03	0,86	0,08	0,86	0,1	0,89	0,09

Tabelle 3: Vergleich der initialen Methoden für den neuen Algorithmus: Mittelwert und Standardabweichung des $mean_{C_L}$ für das Testset.

4.7 Laufzeitvergleich

Der Vergleich der durchschnittlichen Laufzeit in Tabelle 4 über das gesamte Testset zeigt, dass der neue Algorithmus mit der initialen Methode von Kouroggi und fünf Iterationen schneller ist, als die Methode von Kouroggi mit den vorgestellten Optimierungen. Wird die initiale Schätzung durch die Bounding Box durchgeführt, so ist der neue Algorithmus im Mittel schneller als die anderen vorgestellten Verfahren.

Method	mean	std
Kouroggi, Bounding Box+0.25%, IIR, Maxmotion	8,34	7,22
Log-Search, Bounding Box+0.25%, Kouroggi, IIR, Translation-Warp	5,53	2,88
Log-Search, Bounding Box+0.25%, Kouroggi, IIR, Trans-Scal-Rot-Warp, Initial Transformation	6,59	3,8
Log-Search, Bounding Box+0.25%, Initial Transformation	4,76	2,28
Log-Search, Bounding Box, Initial Transformation	4,84	2,12

Tabelle 4: Laufzeitvergleich: Durchschnitt und Standardabweichung der Laufzeit der Matlab-Implementierung in Sekunden (System: Intel Core 2 Duo 1,86 GHz E6320, 1 GB RAM).

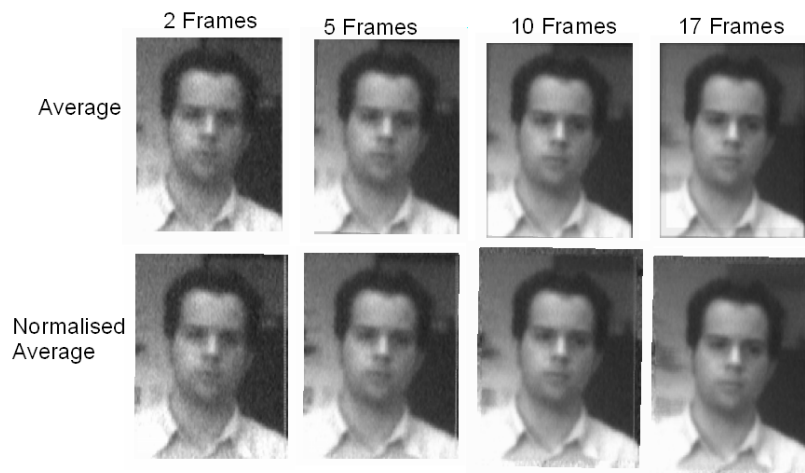


Abbildung 25: Einfluss der Frameanzahl auf das Average Bild

4.8 Superresolution

Die Ergebnisse aus den Experimenten werden in diesem Kapitel eingesetzt um aus einer Sequenz von Bildern ein Superresolution Bild zu erzeugen. Hier stellt sich die Frage: Woran oder wie kann man zeigen, dass das generierte Bild auch ein Superresolution-Bild ist?

Eine erste Möglichkeit ist, das generierte Bild den Bildern der Sequenz gegenüberzustellen und den visuellen Eindruck zu vergleichen. In Abbildung 25 sind aus der verrauschten Sequenz *r002* Average und Normalized Average Bilder für verschiedene Sequenzlängen gegenübergestellt. Mit zunehmender Framezahl wird der Rauschanteil im Bild geringer und mehr Details sichtbar.

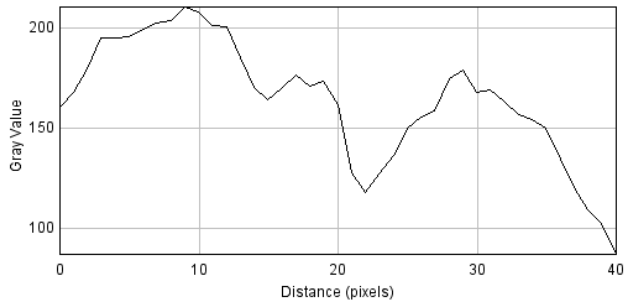
Desweiteren können Profilplots definierter Bereiche einen Anhaltspunkt zur Glättung der Bilder geben. In Abbildung 26 sind die $2\times$ skalierten Average Bilder¹³ dem Referenzbild der Sequenz *n009a* gegenübergestellt.

Wenn die Average-Bilder einen ersten Anhaltspunkt für ein geglättetes Superresolution-Bild geben (vgl. Abschnitt 2.8.1), so müsste sich auch zeigen lassen, dass das Average-Bild mehr Informationen bereithält. Die bisherigen Vergleiche stellen dieses nur für Sequenzen mit Rauschen, sonst aber nicht hinreichend dar. Gesichter beinhalten keine normierten Informationen, an de-

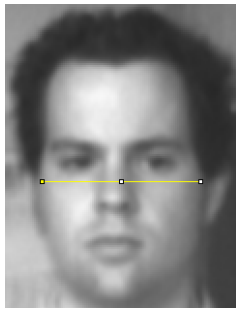
¹³ $2\times$: Die ermittelten Transformationen für die Objektbilder wurden $2\times$ skaliert um die Auflösung der Average Bilder gegenüber dem Referenzbild zu verdoppeln.



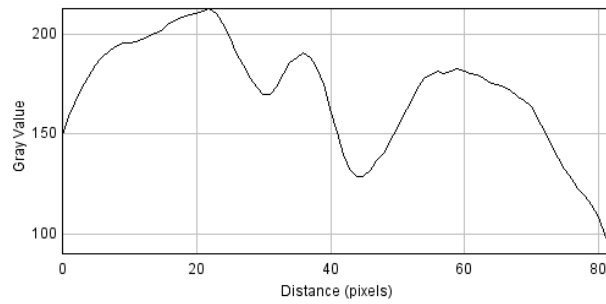
(a) Referenzbild



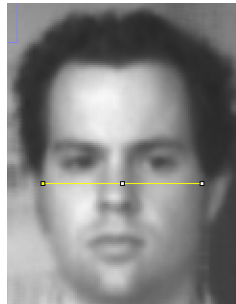
(b) Profilplot des Referenzbildes



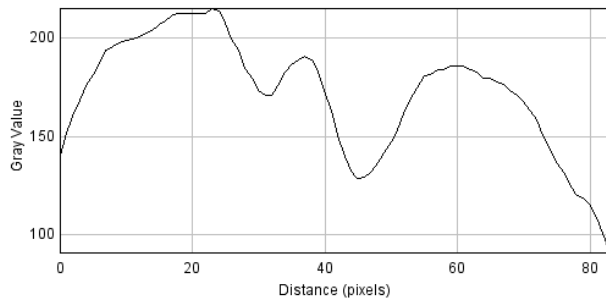
(c) 2×skaliertes Average



(d) Profilplot des Average Bildes



(e) 2×skaliertes Normalize Averaging



(f) Profilplot des Normalized Averaging Bildes

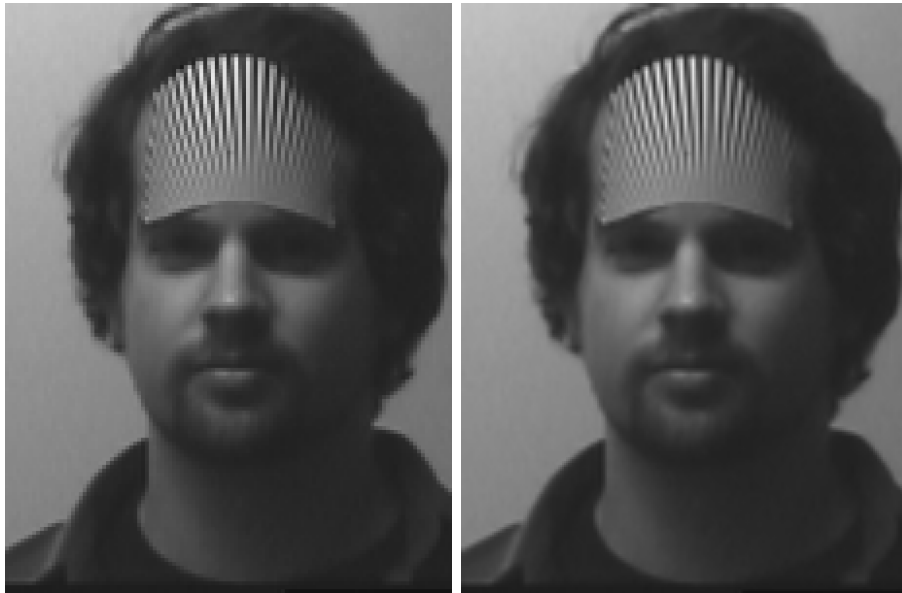
Abbildung 26: Vergleich 2× Average-Verfahren und Referenzbild der Sequenz *n009a* mit 10 Bildern (Registrierung wurde mit dem neuen Algorithmus durchgeführt)

nen sich Superresolution-Verfahren messen lassen. Aus diesem Grund wurden Aufnahmen (siehe z.B. Abb. 45) mit einem Testmuster durchgeführt.

Der visuelle Vergleich in Abbildung 27 zeigt, dass die zuvor verpixelten Linien des Testmusters im Referenzbild, die auch als Artefakte im zweifach mit Interpolation vergrößerten Referenzbild noch zu sehen sind, in den Average-Bildern als durchgehend gerade Linien wahrgenommen werden. Der Vergleich verdeutlicht auch, dass das Average Bild gegenüber dem Normalized Average Bild einen visuell leicht besseren Eindruck hinterlässt. Im Referenzbild (vgl. Abbildung 27 a) der Sequenz ist ein Moiré-Effekt (Aliasing) erkennbar, der durch Nichteinhaltung des Abtasttheorems entstanden ist. Das Raster des CCD-Sensors bzw. die Auflösung der Kamera überlagert sich mit dem Testmuster. Dieses wird durch die gebogenen Interferenzlinien sichtbar. Die Average-Bilder kompensieren diese Artefakte.

Die Abbildung 28 verdeutlicht die Vergrößerung der Auflösung durch das $2\times$ Averaging Verfahren. Durch die gleiche Anzahl an Pixeln wird in dem $2\times$ Average Bild ein kleinerer Ausschnitt des Testmusters wiedergegeben als in dem Referenzbild. Das $2\times$ Average Bild beinhaltet somit mehr Informationen als das Referenzbild.

Die exemplarische Auswertung einer Sequenz durch das *Comparison Window (FaceChecker)* (Abb. 29), in Tabelle 5 zeigt, dass das $2\times$ Average Bild eine höhere Ähnlichkeit mit dem Portaitbild des Probanden erreicht als die anderen Frames der Sequenz.

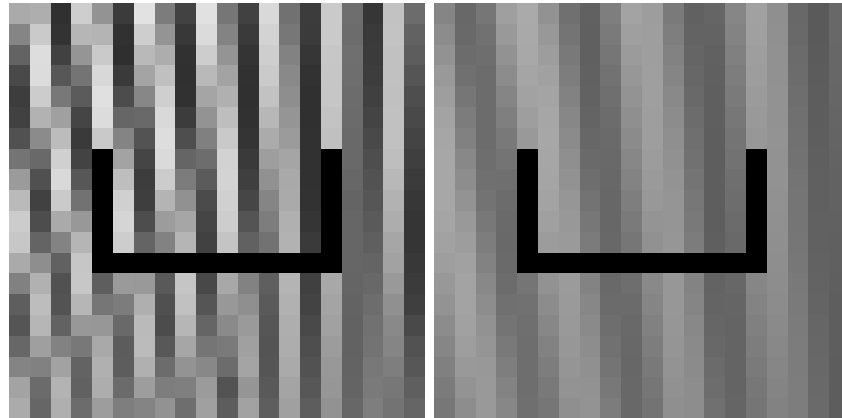


(a) Referenzbild ($2\times$ vergrößert ohne Interpolation) (b) Referenzbild ($2\times$ vergrößert mit Interpolation)



(c) Normalized Average $2\times$ (d) Average $2\times$

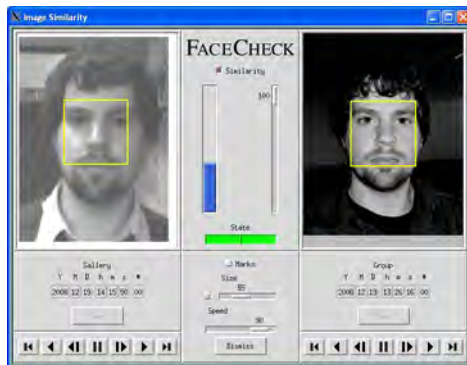
Abbildung 27: Averaging-Verfahren mit Testmuster (Sequenz *sr016b*): Die Bilder zeigen das Referenzbild der Sequenz in Gegenüberstellung zu den Average Bildern. Das Testmuster verdeutlicht, dass die $2\times$ skalierten Average Bilder eine höhere Auflösung aufweisen, als das Referenzbild. (Das Referenzbild wurde nachträglich $2\times$ ohne und mit Interpolation skaliert. Die Interferenzlinien sind auch im Ursprungsbild vorhanden und kein Resultat der Skalierung.)



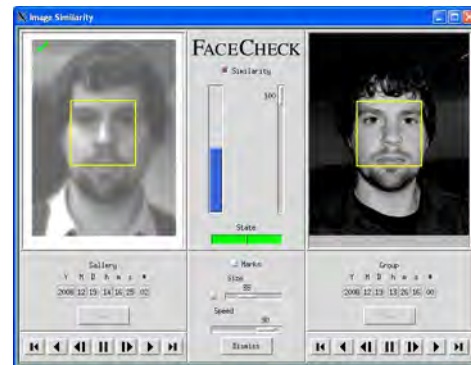
(a) Referenzbild

(b) $2\times$ Average

Abbildung 28: Pixelvergleich Referenzbild - $2\times$ Average Bild: Links ist das Referenzbild und rechts das $2\times$ Average Bild der Sequenz *sr016b* zu sehen. In den Bildern ist jeweils ein Ausschnitt von 10 Pixeln der gleichen Position im Testmuster abgebildet. Das Average Bild bildet mit 10 Pixeln zwei schwarze Linien und das Referenzbild circa 4 schwarze Linien des Testmusters ab.



(a) Ein Bild der Sequenz



(b) $2\times$ Average Bild

Abbildung 29: Ähnlichkeitsvergleich durch das *Comparison Window* des *FaceCheckers*: Links ist ein Bild der Sequenz zu sehen und rechts das Portraitbild der Testperson. Zwischen den Bildern stellt ein blauer Balken die Ähnlichkeit der Gesichter dar. Das $2\times$ Average Bild erreicht eine höhere Ähnlichkeit als alle anderen Bilder der Sequenz.

Image	Similarity
1	0,382
2	0,488
3	0,436
4	0,436
5	0,424
6	0,385
7	0,348
Average	0,503
2×Average	0,507

Tabelle 5: Verbesserung der Gesichtswiedererkennung durch die Average Bilder: Die Tabelle zeigt die erzielte Ähnlichkeit der Einzelbilder der Sequenz und der Average Bilder mit dem Potraitbild des Probanden. Das Average Bild und 2× skalierte Average Bild wird den Einzelbildern der Sequenz gegenübergestellt. Die Average Bilder verbessern die Wiedererkennung des Probanden.

5 Zusammenfassung

In der Arbeit wurden schnelle Verfahren zur Objektregistrierung vorgestellt, die zur Gesichtsstabilisierung verwendet wurden. Die Registrierung wird in einem Ausschnitt des Bildes (Gesichtsregion) durchgeführt, der den relevanten Bereich für die Registrierung eingrenzt. Das Verfahren von Kouroggi [5] wurde durch eine Ausreißer-Verbesserung und Intensitätsanpassung optimiert und einem neuen Algorithmus von Wolfgang Konen [9] gegenübergestellt.

Für die Auswertung der Verfahren wurden umfangreiche Testaufnahmen durchgeführt. Die Untersuchung der Verfahren anhand der Testaufnahmen hat gezeigt, dass die intensitätsbasierte Registrierung [5] auf Grundlage von einzelnen Pixeln in einer Region, die eine grobe Schätzung für den registrierungsrelevanten Bereich darstellt, durch die Optimierungen verbessert werden kann. Das Verfahren ist jedoch nicht robust gegenüber Rauschen und Intensitätsdifferenzen. Der neue Algorithmus [9] erreicht durch die Suche mittels normalisierter Kreuzkorrelation und Logarithmic Search gute Resultate.

Für den Vergleich der registrierten Bilder wurden verschiedene Vergleichsmaße entwickelt, deren Parameter durch systematische Variation bestmöglich ermittelt und auf umfangreichem Testmaterial untereinander und gemäß ihres visuellen Eindrucks verglichen wurden. Ein neues Vergleichsmaß auf Basis des normierten Korrelationskoeffizienten, gibt den visuellen Eindruck der Registrierung gut wieder.

Unter anderem wurden zwei Tools entwickelt: Der *pm_log_inspector*, mit dem die Arbeitsweise des neuen Algorithmus [9] visuell dargestellt werden kann und ein Editor um die Bounding Boxen der Bilder zu editieren.

Mit dem neuen Algorithmus [9] und Averaging-Verfahren wurde gezeigt, dass der neue Algorithmus geeignet ist um Superresolution-Bilder zu erzeugen und die Laufzeit der Matlab-Implementierung im Durchschnitt besser ist, als die des zuvor optimierten Verfahrens von Kouroggi [5].

Aus einer Sequenzen wurde durch die Registrierung mit dem neuen Algorithmus [9] ein Average Bild erzeugt und in der Gesichtserkennung den einzelnen Bildern der Sequenz gegenübergestellt. Hier hat sich gezeigt, dass die Erkennungsleistung durch das stabilisierte Average Bild deutlich verbes-

sert wird.

6 Ausblick

Im Rahmen der Arbeit konnten nicht alle Themen weitreichend recherchiert und ausgearbeitet werden. Weitere Untersuchungen könnten sich mit der Parametrisierung des neuen Algorithmus auseinandersetzen, die hier nicht betrachtet wurden. Für den neuen Algorithmus bieten sich Optimierungsmöglichkeiten an, wie in der Arbeit von Astrid Lundmark [11] erläutert.

Die Echtzeitfähigkeit fand keine Betrachtung, hier wäre interessant, ob sich der ermittelte Geschwindigkeitsvorteil der Matlab-Implementierung auch in einer Java- oder C-Implementierung bewahrheitet.

Die Averaging-Verfahren bieten einen ersten Einstieg in das weite Thema der Superresolution, welches nicht in aller Tiefe diskutiert und erarbeitet werden konnte. Hier bieten sich zahlreiche Verfahren an, angefangen bei dem adaptiven Filter von Tuan Q. Pham und Lucas J. van Vliet[13], der auf dem Normalized Averaging aufsetzt, bis hin zu speziell für Gesichter optimierten Verfahren, wie z.B. die in der Arbeit von David Capel [3] erläuterten.

Die Auswertung der Averaging-Bilder wurde nur exemplarisch anhand des FaceCheckers durchgeführt. Weitreichende Tests mit den Averaging- und anderen Superresolution-Methoden könnten zeigen, dass die Gesichtserkennung durch die Gesichtsstabilisierung optimiert werden kann.

7 Persönliches Schlusswort

Die Bearbeitung des Themas war für mich eine große Herausforderung und von persönlichen Tiefs und Hochs begleitet. Dennoch hat mir die Herausforderung Spaß gemacht, da ich Freude daran hatte neue Ideen auszuprobieren und zu evaluieren. Leider war im Rahmen der Arbeit sowohl zeitlich, als auch inhaltlich die Bearbeitung des Themas Superresolution nicht weitreichend möglich, was ich schade finde, da gerade das einen Wow-Effekt erzielt.

Die Arbeit hat für mich persönlich zu einem sehr zufriedenstellenden Re-

sultat geführt, geht man von dem anfänglich kaum einsetzbaren Verfahren aus.

Das Thema der Gesichtserkennung beziehungsweise dessen Einsatzgebiet hat für mich persönlich einen negativen Beigeschmack. Ich hoffe, dass einige Ergebnisse in das Forschungsprojekt *Bildverarbeitung und 3D-Navigation in der Endoskopie* von Wolfgang Konen einfließen, und somit auch in einem ganz anderen Kontext eine wichtige Bedeutung erfahren.

Literatur

- [1] o.V., o.J.: Bildregistrierung, <http://de.wikipedia.org/wiki/Bildregistrierung>, 06.12.2008.
- [2] Rat der Europäischen Union: VERORDNUNG (EG) Nr. 2252/2004 DES RATES, 13. Dezember 2004.
- [3] David Capel: Image Mosaicing and Super-resolution, Springer-Verlag, London, 2004.
- [4] Jürgen Fuchs: Warum die Bildqualität der Überwachungsanlagen den Ermittler nicht immer begeistert, Polizei heute 3/2007, Richard Boorberg Verlag, S. 119-121, 2007.
- [5] Masakatsu Kouroggi et. al.: Real-Time Image Mosaicing From a Video Sequence., In: Procs ICIP99, vol.4, S. 133-137, 1999.
- [6] Wolfgang Konen et al.: Endoscopic image mosaics for real-time color video sequences, In: H.U. Lemke (ed.), Computer Assisted Radiology and Surgery (CARS2007, Berlin), Elsevier, Amsterdam, 2007.
- [7] Wilhelm Burger und Mark J. Burge: Digitale Bildverarbeitung: Eine Einführung mit Java und ImageJ, Auflage 2, Springer-Verlag, Berlin, Juni 2006.
- [8] Scott Shald: Comparison of FIR and IIR Filters in Coherent Lidar Processing, 14th Coherent Laser Radar Conference, Snowmas, Colorado, USA, August 2007
- [9] Wolfgang Konen: persönliche Mitteilung, Dezember 2008.
- [10] J.R. Jain and A.K. Jain: Displacement measurement and its application in interframe image coding, IEEE Transactions on Communications, vol. COM-29, S. 1799-1808, Dec 1981.
- [11] Astrid Lundmark: Non-Redundant Search Patterns in Log-Search Motion Estimation, SSAB Symposium on Image Analysis, März 2001.

- [12] o.V., o.J.: CCD-Sensor, <http://de.wikipedia.org/wiki/CCD-Sensor>, 06.12.2008.
- [13] T.Q. Pham and L.J. van Vliet: Super-resolution Fusion using Adaptive Normalized Averaging, in: J.J. van Wijk, J.W.J. Heijnsdijk, K.G. Langendoen, R. Veltkamp (eds.), ASCI 2004; Proceedings of the tenth annual conference of the Advanced School for Computing and Imaging (Ouddorp, NL, June 2-4), ASCI, Delft, 2004, 107-114
- [14] CompuServe Incorporated: Graphics Interchange Format - A standard defining a mechanism for the storage and transmission of raster-based graphics information, Columbus, Ohio, 07.1987, <http://www.w3.org/Graphics/GIF/spec-gif87.txt>
- [15] CompuServe Incorporated: Graphics Interchange Format Version 89a, Columbus, Ohio 31.07.1990 <http://www.w3.org/Graphics/GIF/spec-gif89a.txt>
- [16] o.V., o.J.: High Resolution Test Pattern, <http://www.bealecorner.org/red/test-patterns/star-chart-bars144-600dpi.png>, 09.12.2008.

8 Anhang

A FaceSnapRecorder

Das System der Firma Crossmatch Technologies, mit dem die Gesichtsaufnahmen für diese Arbeit gemacht wurden, ist eine Software zur Gesichtserkennung (*FaceSnap*) und Wiedererkennung (*FaceCheck*). Es umfasst einen Rechner, der mit einer Framegrabber Karte ausgestattet ist. Das Betriebssystem, welches die Grundlage des Systems bildet, basiert auf einem angepassten Linux, welches nach dem Booten in einem so genannten Kiosk-Mode startet. Das heißt man hat, bis auf die Möglichkeit, mittels der Tastenkombination *ALT+F1* bis *F4* (grafische Oberfläche) auf die Konsolen zu wechseln, keinerlei Möglichkeiten auf das Betriebssystem Einfluss zu nehmen. Die erste Konsole *ALT+F1* wird hierbei für das starten der X-Window-Oberfläche und des FaceSnapRecorders (FSR) benutzt und stellt keinen Login-Prompt zur Verfügung. Desweiteren ist das System mit einer Netzwerkkarte und einem CD-Brenner ausgestattet. Über diese Schnittstellen ist es möglich auf die erzeugten Gesichtsaufnahmen zuzugreifen, indem man Bilder auf CD brennt oder über das Netzwerk mittels SMB auf die Shares des Samba-Server zugreift. Im Folgenden sind Eingaben und Widgets¹⁴ durch *hervorgehobene* Schrift gekennzeichnet.

Inbetriebnahme

Hardware

Um das System in Betrieb zu nehmen, benötigt man eine Kamera, die über einen der drei RCA¹⁵ oder den SVHS-Eingang an die Framegrabber-Karte angeschlossen wird, eine Tastatur und eine Maus mit PS/2 Steckern, sowie einen Monitor, der über einen Sub-D Stecker verfügt.

¹⁴Element einer grafischen Benutzer-Oberfläche

¹⁵Steckverbindung für Composite Video

Erster Start

Das System startet mit dem *FaceSnapRecorder*. Als erstes muss man die Oberfläche mit einem Passwort freischalten. Dieses ist im Auslieferungszustand: *0000*. Nun untergliedert sich die Oberfläche in Widgets, die sich auf den Modus beziehen und welche, die den Modus wechseln, sowie eine *Tooltip-Leiste (4)*, die sich am unteren Rand befindet und zu den anderen Widgets wie z.B. Buttons eine kurze Funktionsbeschreibung liefert. Das System gliedert funktional folgendermaßen:

- FaceSnapRecorder
 - *Record (9)*
 - *Replay (7)*

- FaceChecker
 - *Search (6)*
 - *Train (5)*
 - *FaceCheck*

Hieraus resultieren zwei Betriebsarten:

FaceSnapRecorder: Fortlaufende Erkennung und Aufzeichnung von Gesichtern aus einer Videoquelle

FaceChecker: Fortlaufende Gesichtswiedererkennung

Konfiguration

Es gibt zahlreiche Konfigurationsmöglichkeiten, von denen an dieser Stelle jedoch nur die erwähnt werden, die für die Aufnahme von Bildsequenzen relevant sind. Um Gesichter aufzunehmen, ist es als erstes nötig, die Kamera auf die Szene zu kalibrieren. Im Modus *Record (9)* ist über *Setup→Image Acquisition* die Einstellung bezüglich der Kamera und der Aufnahmeparameter möglich.

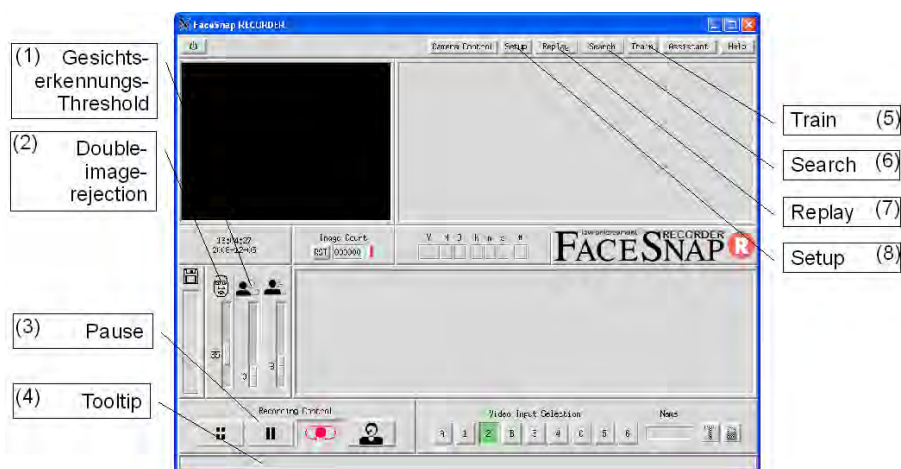


Abbildung 30: Betriebsart FaceSnapRecorder

Image Acquisition

Zuerst wählt man über Video Input den Video-Kanal, der je nach ausgesuchtem Anschluss an der Framegrabber-Karte variiert. Weitere wichtige Einstellungsparameter sind das Video Format, sowie die Einstellungen zum Gesicht unter dem Reiter *Face*. Hier sollte man die *Thresholds* (1) für die Gesichtserkennung und die *double image rejection* (2) einstellen sowie den Blickwinkel der Kamera auf das Gesicht auswählen. Weitere sehr wichtige Parameter betreffen die minimale, sowie die maximale Gesichtsgröße (10), wobei das Minimum auf eine Größe von 25 Pixeln begrenzt ist. Um Regionen aus der Erkennung auszuschließen, kann man eine *region of no interest* (11) in dem Standbild auswählen (Abb. 31).

Desweiteren gibt es die Möglichkeit Helligkeit, Kontrast und Intensität des Bildes einzustellen, dieses ist entweder manuell über die Schieberegler oder automatisch unter dem Reiter *Auto Brightness Adjustment* über den Button *Auto ADC* zu erreichen (Abb. 32).

Passwort

Es gibt zwei Passworte, die für die Benutzung des Systems relevant sind. Zum einen das Passwort für die Benutzeroberfläche und ein weiteres für den Zugriff auf den Samba-Server. Im Auslieferungszustand sind diese folgendermaßen:

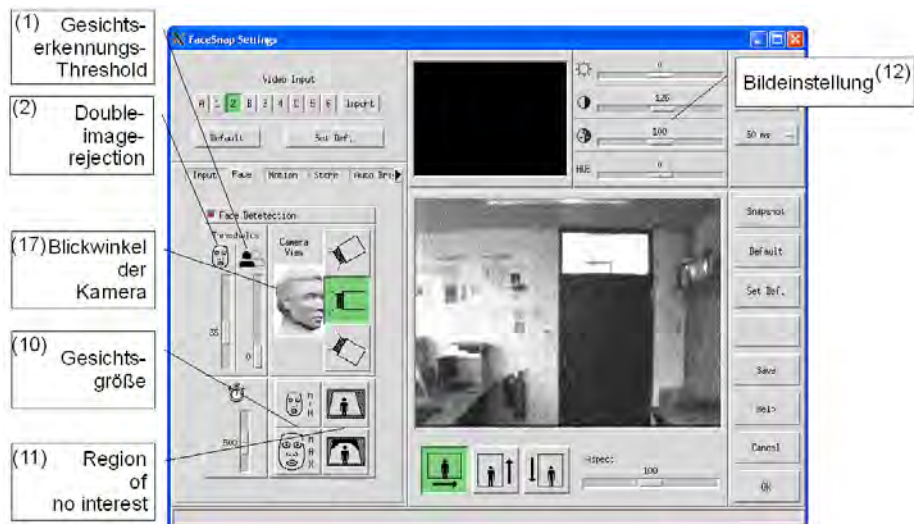


Abbildung 31: Einstellung der Aufnahmeparameter für die Gesichtserkennung

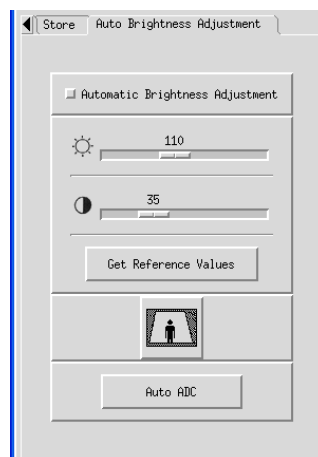


Abbildung 32: Helligkeitseinstellungen

Userinterface: 0000

Samba: user: fsr, password: 0000

Das Passwort für das Userinterface ist im Modus *Record (9)* unter dem Menüpunkt *Setup*→*Security* änderbar. Die Änderung des Passworts für Authentifizierung am Samba Server ist im Abschnitt A erläutert.

Betrieb

Im Nachfolgenden wird ein einfaches Szenario dargestellt, welches die Bedienung der Software verdeutlichen soll. Zunächst nimmt man im Modus *Record (9)* einige Gesichter auf, hierbei ist wichtig, dass im Widget *Recording Control* der *Pause-Button (3)* nicht aktiviert ist.

Im Modus *Replay (7)* kann man nun die aufgenommenen Bilder in zeitlicher Reihenfolge betrachten und evtl. unerwünschte löschen oder Zeitfenster untersuchen. Zu beachten ist, dass man nur die Aufnahmen eines Volumens betrachten kann. Diese kann man unter dem Menüpunkt *Options*→*Work Volume* auswählen, wobei im *Volume A* alle Aufnahmen gespeichert sind und im *Volume B* zum Beispiel die Watchlist abgelegt ist.

Man hat zwei Möglichkeiten ein Zeitfenster auszuwählen. Die erste Möglichkeit ist das Intervall genau über Start- und Endzeitpunkt zu definieren. Die zweite ist, in dem *Recording-Histogram (15)* das Intervall per Maus zu bestimmen, indem man mit dem Mauscursor auf die gewünschte Startposition geht und nun die mittlere Maustaste gedrückt hält, den Mauscursor bis zum gewünschten Ende fährt und die mittlere Maustaste wieder los lässt. Der Zoom kann mittels rechter Maustaste auf dem Histogramm rückgängig gemacht werden, wobei mit jedem Klick die vorherige Auswahl-Zoomstufe zurückgenommen wird.

Für das Löschen selektiert man die gewünschten Bilder und markiert sie über die *markier Tools (10)*, danach klickt man auf den *Mülleimer (12)*. Um nun mit den Gesichts-Aufnahmen eine Wiedererkennung durchzuführen, muss man in den Modus *Search (6)* wechseln. Hier kann man eine Gruppe von maximal 11 Bildern erstellen, die jedoch mindestens 5 Bilder für gute Ergebnisse enthalten sollten. Diese kann entweder verschiedene oder aber

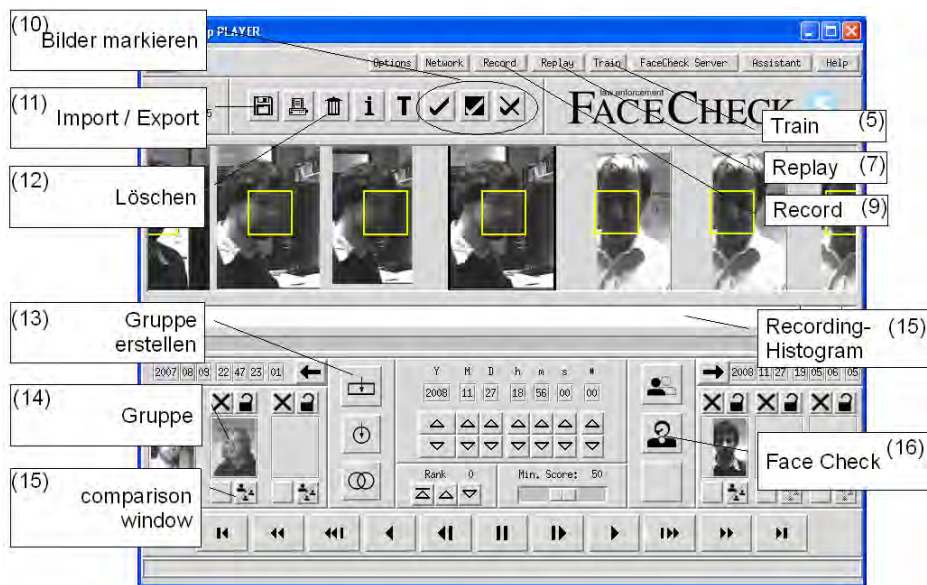


Abbildung 33: Search-modus zum Betrachten und Verwalten von Aufnahmen

auch nur eine Person wiedergeben. Die Ähnlichkeit der Gesichter der Gruppe im Vergleich zur Gallerie kann über den Button *Open image comparision window* (15) durchgeführt werden (Abb. 34). Die genauen Ähnlichkeitswerte können auf der ersten Konsole als debug-Ausgaben abgelesen werden.

Die selektierten Bilder können dann über *Train*→*Group with one person* oder *Train*→*Group with different persons* trainiert werden. Um gute Ergebnisse zu erreichen, sollten mindestens 10 Personen trainiert werden, bevor man nach Personen über die Watchlist sucht. Nun ist es möglich Personen zu Watchlist hinzuzufügen, indem man mehrere Bilder einer Person markiert und diese dann über *Assistant*→*Takeover selected images of person into watchlist* zur Watchlist hinzufügt.

Mit den aktiven Personen der Watchlist kann man nun im Modus *Search* (6) die gesamte Gallerie durchsuchen lassen, indem man auf den Button *Search for images which match active classes* (16) klickt. Die Bilder die den Personen auf der Watchlist entsprechen, werden durch eine Markierung hervorgehoben. Im laufenden Betrieb kann über *Assistant*→*Start FaceCheck Continous Operation* das Videosignal überwacht werden.

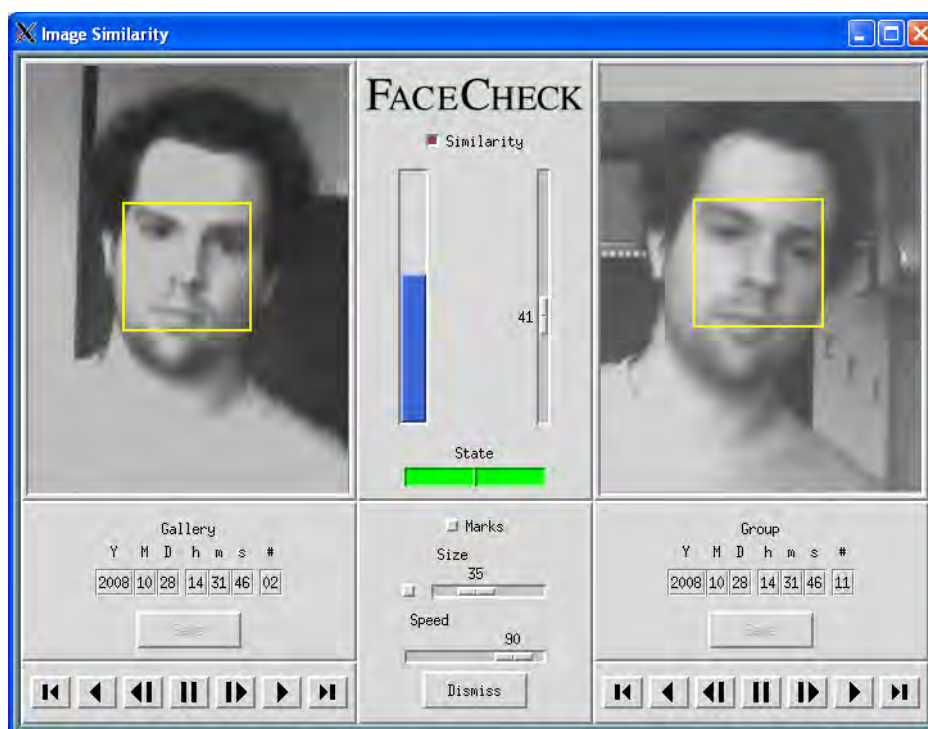


Abbildung 34: Vergleich verschiedener Gesichter einer Gruppe mit der Galerie

Export

Der Zugriff auf die aufgezeichneten Bilder ist über verschiedene Schnittstellen möglich:

- Netzwerk
- Diskette
- CD-Rom

Um über das Netzwerk auf die Daten zugreifen zu können, benötigt man zunächst ein Passwort, das im Modus *Replay* über das Menü *Network->Network Settings, Set Password for Windows Access* gesetzt werden kann. Nun kann man mittels eines SMB-Clients wie z.B. dem Explorer von Windows auf das Share des Samba-Servers zugreifen. Hierfür benötigt man die IP des Rechners, die auch den *Network Settings* zu entnehmen bzw. einzustellen ist. Um Bilder auf ein anderes Medium zu exportieren muss man zuerst die gewünschten markieren und kann dann anschließend auf den Button *Diskette (11)* klicken und nachfolgend das gewünschte Zielmedium und Datenformat auswählen.

Import

Der Import funktioniert analog zum Export, nur das hierfür zuvor keine Bilder in der Galerie markiert werden müssen. Als Medium bietet sich hier das *Clipboard* an, auf welches über das Samba-Share zugegriffen werden kann. Man kopiert mittels eines SMB-Clients die gewünschten Bilder in das Verzeichnis *Clipboard* im Samba-Share und führt den Import durch. Die importierten Bilder befinden sich hiernach in der Galerie.

B Bounding Box im Gif Kommentar

Jedes Gesicht, das durch den FaceSnapRecorder erkannt wird, wird im GIF-Format mit zusätzlichen Informationen im Kommentarblock des Bildes abgespeichert. Der Kommentar ist durch key-value Paare gekennzeichnet, die durch Zeilenumbrüche voneinander getrennt sind. Der für die Gesichts Position und Größe verantwortliche Eintrag ist *boundary=* gefolgt von 8 Integer Werten getrennt durch Leerzeichen. Dies sind die vier Koordinaten der Gesichtsregion, die jeweils durch ihre x und y Werte repräsentiert sind. Um den Kommentar-Block aus einer Gif-Datei auslesen zu können, muss man die Datei parsen, da es keine fest definierte Position für den Kommentar gibt.

B.1 GIF-Format

Das Gif-Format [14][15] wird durch einen Header eingeleitet, der die Version des Formates definiert. Für die Kommentar-Extention sollte dies *GIF89a* sein. Der Kommentarblock kann auch in Version *GIF87* vorhanden sein, wird aber von den Parsern als unbekannter Erweiterungs-Block ignoriert.

Hierauf folgt der *Screen Descriptor*, der die Auflösung des Bildes und eine optionale globale Farbtabelle einleitet. Nachfolgend unterteilt sich der Datenstrom in Extention- und Imageblöcke. Der Kommentar ist in einem Extention-Block abgespeichert. Die eigentlichen Daten innerhalb eines Image- oder Extention-Abschnittes sind durch Daten-Blöcke, bestehend aus Subblöcken abgespeichert. Jeder Subblock, hat eine maximale Größe von 255 Byte:

	7 6 5 4 3 2 1 0	Field Name	Type
	+-----+		
0		Block Size	Byte
	+-----+		
1			
	+- -+		
2			
	+- -+		
3			
	+- -+		
		Data Values	Byte
	+- -+		
up			
	+- -+		
to			
	+- -+		

```

      +-                +-
255 |                  |
      +-----+

```

[Quelle: CompuServe Incorporated: GRAPHICS INTERCHANGE FORMAT Version 89a]

Ist das Ende des letzten Subblocks erreicht, wird dieses durch einen Block-Terminator indiziert, welches einer Block Size von Null entspricht:

```

      7 6 5 4 3 2 1 0      Field Name      Type
      +-----+
0 |                  |      Block Size      Byte
      +-----+

```

[Quelle: CompuServe Incorporated: GRAPHICS INTERCHANGE FORMAT Version 89a]

In Tabelle 6 ist das Gif89a Format schematisch dargestellt. Der Sourcecode einer Java-Implementierung zum parsen der Gif-Dateien ist auf der beigelegten DVD zu finden.

B.2 Bounding Box Editor

Der Bounding Box Editor (Abb. 35) ist ein Tool mit dem man die Bounding Box der Gif-Dateien bearbeiten kann. Er ermöglicht die Kontrolle, Korrektur und ein nachträgliches Einfügen einer Bounding Box. Die Matlab-Implementierung der Algorithmen ist somit unabhängig von dem FaceSnapRecorder, da in eigene Aufnahmen nachträglich eine Bounding Box eingefügt werden kann.

Der Editor startet mit einem Datei-Öffnen-Dialog, indem eine Gif-Datei zur Bearbeitung ausgewählt wird. Über die Buttons *previous* und *next* kann in dem Verzeichnis, (in dem sich die geöffnete Datei befindet), das vorherige oder nächste Bild geöffnet werden, wodurch man einen direkten Überblick über die Bounding Box in der Sequenz bekommt. Es gibt zwei Möglichkeiten, die Bounding Box zu bearbeiten. Zum einen über die Texteingabefelder für den Punkt *P1* (oben-links) und den Punkt *P2* (unten-rechts) oder per Mauscursor direkt in dem Bild. Hierfür fährt man mit dem Mauscursor zur gewünschten Position für die linke obere Ecke (*P1*) der Box und drückt die linke Maustaste. Diese hält man gedrückt und fährt mit dem Mauscursor zur gewünschten Position für die rechte untere Ecke (*P2*) der Bounding Box. Die Änderung kann man über den Button *save* in die Datei schreiben.

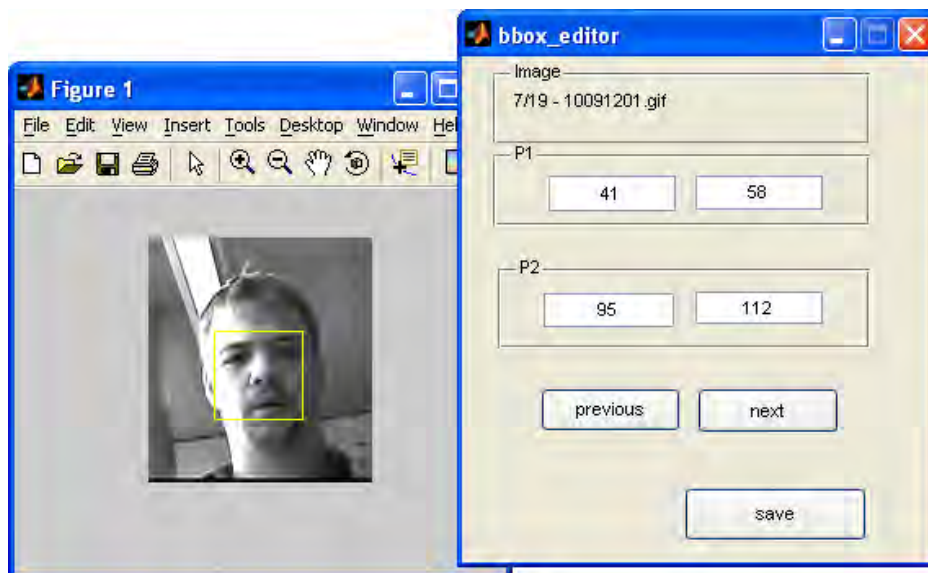


Abbildung 35: Boundig Box Editor

Bytes	Typ	Beschreibung
6	Header	Version (für Kommentar Extension mindestens GIF89a)
7	Screen Descriptor	Globale Eigenschaften wie z.B. Auflösung, 4.Byte wichtig: MSB-Flag signalisiert das eine globale Farbtabelle folgt und die drei LSBs geben die Größe der Tabelle in Byte an: $3 \cdot 2^{LSBs+1}$
	Global Color Table	Farbzuordnungen, Größe über Screen Descriptor definiert
...		
1	Image Separator (0x2C)	Leitet einen Imageblock ein
9	Image Descriptor	Lokale Eigenschaften des Bildes wie z.B. Größe und Position. 8.Byte wichtig: gleiche Funktion wie das 4.Byte beim Screen Descriptor. Bestimmt die Größe der nachfolgenden Local Color Table
	Local Color Table	Lokale Farbtabelle für ein Bild
	LZW-Block	Datenblock bestehend aus LZW komprimierten Subblöcken
...		
1	Extension Introducer (0x21)	Signalisiert einen Gif-Erweiterungs-Block
1	Extension Label	Typ des Extension Blocks
	Daten-Block	Daten Block bestehend aus Subblöcken
...		
1	Extension Introducer (0x21)	Signalisiert einen Gif-Erweiterungs-Block
1	Comment Label (0xFE)	Kommentarblock
	Daten-Block	Datenblock bestehend aus Subblöcken
...		
1	GIF TRAILER (0x3B)	Ende des Gif-Datenstroms

Tabelle 6: Schematische Darstellung des GIF-Formats

C Aufnahmen

Die Aufnahmen wurden mit der Kamera *DMK 4002-IR/C* des Herstellers *IMAGINGSOURCE* und zwei verschiedenen Objektiven durchgeführt: *Pentax 3.5-8mm* und einem *Nikon-S 70-210mm*. Die Aufnahmen für das Super-resolution Experiment wurden mit dem *Nikon-S 70-210mm* Objektiv durchgeführt.

IMAGINGSOURCE DMK 4002-IR/C	
Video-Format	CCIR
Typ	CCD Interline Transfer, EXview HAD
Format	1/2"
Auflösung	752 × 582
Pixel-Größe	H:8,6 μ m, V:8,3 μ m
Video aus	Composite Video

Tabelle 7: Auszug aus den technischen Daten der Kamera

Die Aufnahmen unterteilen sich in drei Kategorien:

1. Gleichbleibendem Lichteinfall (gekennzeichnet durch ein n im Verzeichnisnamen)
2. Änderung der Lichtverhältnisse (gekennzeichnet durch ein b im Verzeichnisnamen)
3. Erhöhter Rausch-Anteil (gekennzeichnet durch ein r im Verzeichnisnamen)

Mit dem Objektiv *Nikon-S 70-210mm* wurden Aufnahmen mit Ausschnitten eines Testmusters [16] durchgeführt (gekennzeichnet durch ein sr im Verzeichnisnamen). In Tabelle 8 sind alle Sequenzen mit Beschreibung aufgeführt, gefolgt von einem Auszug einzelner Frames. Die vollständigen Testsets sind auf der beigelegten DVD zu finden.

Seq.	Frames	Objektiv	Bounding Box Größen in Pixeln	Transformation	Beleuchtung	Rauschen	Testmuster	Beschreibung
b001	18	Pentax 3.4-8mm	32 - 36	geringe Translation	seitlich, starke Variation	-	-	Person ändert Mimik unter starker Beleuchtungsvariation
b002	11	Pentax 3.4-8mm	36 - 49	Skalierung, geringe Translation	seitlich, geringe Variation	-	-	Person nähert sich der Kamera
b003	22	Pentax 3.4-8mm	32 - 43	geringe Rotation, geringe Skalierung, geringe Translation	seitlich, starke Variation	-	-	Person nähert sich der Kamera und neigt hierbei leicht das Gesicht
b004	13	Pentax 3.4-8mm	34 - 50	geringe Translation, Skalierung	seitlich, starke Variation, Sättigung	-	-	Person nähert sich der Kamera, die ersten Frames sind überbelichtet, dann wandert ein Schatten durchs Gesicht
b005	19	Pentax 3.4-8mm	33 - 64	Translation, starke Skalierung	seitlich, starke Variation	-	-	seitliche Beleuchtung, Skalierung (Person nähert sich der Kamera)
b006	18	Pentax 3.4-8mm	33 - 67	Translation, starke Skalierung	gleichmäßig, geringe Variation	-	-	Person nähert sich der Kamera
b007	28	Pentax 3.4-8mm	33 - 53	Translation, Skalierung	gleichmäßig, starke Variation	-	-	Person nähert sich seitlich der Kamera
b008	14	Pentax 3.4-8mm	32 - 43	Translation, Skalierung	gleichmäßig, mittlere Variation	-	-	Person nähert sich frontal zur Kamera
b009	17	Pentax 3.4-8mm	58 - 72	geringe Rotation, Translation, geringe Skalierung	seitlich-oben, geringe Variation	-	-	Person nähert sich der Kamera und neigt hierbei leicht das Gesicht
b010	51	Pentax 3.4-8mm	38 - 49	Translation, Skalierung	seitlich-oben, geringe Variation	-	-	Person nähert sich der Kamera und bewegt den Kopf leicht seitlich
b011	24	Pentax 3.4-8mm	32 - 42	geringe Rotation, Translation, geringe Skalierung	seitlich, geringe Variation	-	-	geringe Kopfbewegungen der Person
n001	31	Pentax 3.4-8mm	33 - 41	geringe Rotation, geringe Translation	seitlich	-	-	geringe Kopfbewegungen der Person

Seq.	Frames	Objektiv	Bounding Box Größen in Pixeln	Transformation	Beleuchtung	Rauschen	Testmuster	Beschreibung
n002	35	Pentax 3.4-8mm	32 - 43	geringe Translation	seitlich	-	-	Person ändert Mimik und spricht
n003	19	Pentax 3.4-8mm	47 - 60	geringe Rotation, geringe Translation	seitlich	-	-	geringe Kopfbewegungen der Person
n004	20	Pentax 3.4-8mm	48 - 62	geringe Rotation, geringe Skalierung, geringe Translation	seitlich	-	-	geringe Kopfbewegungen der Person
n005	9	Pentax 3.4-8mm	33 - 43	Rotation, geringe Skalierung, geringe Translation	seitlich	-	-	Person neigt den Kopf
n006	22	Pentax 3.4-8mm	33 - 49	geringe Translation, geringe Skalierung	gleichmäßig	-	-	Person nähert sich der Kamera
n007	8	Pentax 3.4-8mm	32 - 38	Translation, Skalierung	gleichmäßig	-	-	Person entfernt sich von der Kamera
n008	13	Pentax 3.4-8mm	38 - 58	Translation, Skalierung	gleichmäßig	-	-	Person nähert sich der Kamera
n009	32	Pentax 3.4-8mm	32 - 53	Translation, Skalierung	gleichmäßig	gering	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf
n010	22	Pentax 3.4-8mm	70 - 87	Translation	gleichmäßig	-	-	Person steht dich vor der Kamera
n011	13	Pentax 3.4-8mm	33 - 45	geringe Rotation, Translation	gleichmäßig	-	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf
n012	14	Pentax 3.4-8mm	69 - 102	Translation, Skalierung	gleichmäßig	-	-	Person steht dich vor der Kamera, bewegt den Kopf und ändert seine Mimik
n013	14	Pentax 3.4-8mm	33 - 53	geringe Rotation, Translation, Skalierung	gleichmäßig dunkel	-	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf
n014	14	Pentax 3.4-8mm	33 - 38	geringe Translation, Skalierung	gleichmäßig dunkel	-	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf

Seq.	Frames	Objektiv	Bounding Box Größen in Pixeln	Transformation	Beleuchtung	Rauschen	Testmuster	Beschreibung
n015	5	Pentax 3.4-8mm	34 - 39	geringe Translation, geringe Skalierung	gleichmäßig dunkel	-	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf
n016	36	Pentax 3.4-8mm	58 - 89	Translation, Skalierung	gleichmäßig	-	-	Person entfernt sich und bewegt sich wieder auf die Kamera zu, geringe Neigung des Kopfes
n017	16	Pentax 3.4-8mm	48 - 68	geringe Rotation, Translation, Skalierung	gleichmäßig	-	-	Person bewegt sich seitlich zur Kamera und dabei auf die Kamera zu
n018	44	Pentax 3.4-8mm	38 - 52	geringe Translation, Skalierung	gleichmäßig	-	-	Person bewegt sich mit starrem Gesicht auf die Kamera zu
n019	17	Pentax 3.4-8mm	39 - 50	geringe Rotation, Translation, geringe Skalierung	gleichmäßig dunkel	-	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf
n020	14	Pentax 3.4-8mm	76 - 98	geringe Translation	gleichmäßig	-	-	Person steht dicht vor der Kamera
r001	38	Pentax 3.4-8mm	34 - 44	geringe Translation, geringe Skalierung	gleichmäßig	stark	-	Person nähert sich der Kamera
r002	27	Pentax 3.4-8mm	42 - 74	Translation, Skalierung	gleichmäßig	stark	-	Person nähert sich der Kamera
r003	10	Pentax 3.4-8mm	94 - 113	Translation, geringe Skalierung	gleichmäßig dunkel	stark	-	Person steht dicht vor der Kamera
r004	20	Pentax 3.4-8mm	57 - 73	Translation	gleichmäßig dunkel	mittel	-	Person bewegt nur geringfügig den Kopf
r005	19	Pentax 3.4-8mm	37 - 77	Translation, Skalierung	seitlich-frontal, Variation	stark	-	Person nähert sich der Kamera
r006	9	Pentax 3.4-8mm	33 - 50	geringe Rotation, Translation, Skalierung	gleichmäßig	mittel	-	Person nähert sich der Kamera zu und neigt dabei leicht den Kopf
r007	7	Pentax 3.4-8mm	68 - 78	Translation, geringe Skalierung	dunkel, geringe Variation	leicht	-	Person neigt den Kopf

Seq.	Frames	Objektiv	Bounding Box Größen in Pixeln	Transformation	Beleuchtung	Rauschen	Testmuster	Beschreibung
r008	12	Pentax 3.4-8mm	77 - 94	Translation	geringe Variation	leicht	-	Person neigt den Kopf
r009	16	Pentax 3.4-8mm	73 - 101	Translation	starke Variation	stark	-	Person neigt den Kopf
r010	17	Pentax 3.4-8mm	51 - 103	Translation, Skalierung	starke Variation	leicht	-	Person entfernt sich von der Kamera
r011	35	Pentax 3.4-8mm	32 - 41	geringe Translation, geringe Skalierung	gleichmäßig	leicht	-	Person neigt den Kopf, geringe Bewegung
r012	15	Pentax 3.4-8mm	39 - 61	Translation, Skalierung	geringe Variation	mittel	-	Person nähert sich der Kamera
r013	11	Pentax 3.4-8mm	36 - 65	Translation, Skalierung	Variation	stark	-	Person nähert sich der Kamera
r014	13	Pentax 3.4-8mm	59 - 69	geringe Rotation, Translation, Skalierung	seitlich-oben, Variation	leicht	-	Person entfernt sich von der Kamera und neigt dabei leicht den Kopf
r015	15	Pentax 3.4-8mm	102 - 122	Translation, geringe Skalierung	gleichmäßig	leicht	-	Person bewegt sich seitlich zur Kamera
r016	10	Pentax 3.4-8mm	40 - 46	geringe Translation, geringe Skalierung	gleichmäßig	leicht	-	Person entfernt sich von der Kamera
r017	23	Pentax 3.4-8mm	38 - 47	geringe Rotation, geringe Translation, geringe Skalierung	geringe Variation	leicht	-	Person nähert sich der Kamera
sr001	9	Nikon S Tele 70-210mm	85 - 96	Translation, geringe Skalierung	geringe Variation	-	grob	Person entfernt sich von der Kamera, Aufnahmen leicht geblurred
sr002	11	Nikon S Tele 70-210mm	47 - 55	geringe Translation, geringe Skalierung	seitlich-oben, Variation	-	fein	Person läuft unter Lichtquelle durch

Seq.	Frames	Objektiv	Bounding Box Größen in Pixeln	Transformation	Beleuchtung	Rauschen	Testmuster	Beschreibung
sr003	75	Nikon S Tele 70- 210mm	69 - 94	geringe Translation, geringe Skalierung	gleichmäßig	-	sehr fein	Person nähert sich der Kamera
sr004	15	Nikon S Tele 70- 210mm	90 - 98	geringe Translation	Variation	-	sehr fein	Person nähert sich der Kamera und bewegt nur geringfügig den Kopf
sr005	31	Nikon S Tele 70- 210mm	68 - 85	geringe Translation, geringe Skalierung	geringe Variation	-	sehr fein	Person nähert sich der Kamera und bewegt nur geringfügig den Kopf
sr006	36	Nikon S Tele 70- 210mm	78 - 102	geringe Translation, geringe Skalierung	geringe Variation	-	sehr fein	Person nähert sich der Kamera und bewegt nur geringfügig den Kopf
sr007	39	Nikon S Tele 70- 210mm	77 - 100	geringe Translation, geringe Skalierung	geringe Variation	-	sehr fein	Person nähert sich der Kamera und bewegt nur geringfügig den Kopf
sr008	136	Nikon S Tele 70- 210mm	77 - 102	geringe Translation, geringe Skalierung	geringe Variation	-	fein	Person nähert sich der Kamera und bewegt nur geringfügig den Kopf
sr009	100	Nikon S Tele 70- 210mm	77 - 102	geringe Translation, geringe Skalierung	geringe Variation	-	fein	Person nähert sich der Kamera und bewegt nur geringfügig den Kopf
sr010	60	Nikon S Tele 70- 210mm	77 - 103	geringe Translation, geringe Skalierung	geringe Variation	-	sehr fein	Person entfernt sich von der Kamera
sr011	21	Nikon S Tele 70- 210mm	80 - 97	geringe Translation, geringe Skalierung	Variation	-	grob	Person nähert sich der Kamera

Seq.	Frames	Objektiv	Bounding Box Größen in Pixeln	Transformation	Beleuchtung	Rauschen	Testmuster	Beschreibung
sr012	17	Nikon S Tele 70- 210mm	52 - 63	geringe Translation, geringe Skalierung	dunkel, starke Variation	-	fein	Person bewegt sich geringfügig auf die Kamera zu
sr013	30	Nikon S Tele 70- 210mm	44 - 53	geringe Translation, geringe Skalierung	geringe Variation	-	fein	Person bewegt nur geringfügig den Kopf, Moiré-Effekt und Treppchenbildung im Testmuster erkennbar
sr014	12	Nikon S Tele 70- 210mm	39 - 48	geringe Translation, geringe Skalierung	dunkel, Variation	-	fein	Person bewegt nur geringfügig den Kopf, Treppchenbildung im Testmuster erkennbar
sr015	16	Nikon S Tele 70- 210mm	46 - 57	geringe Translation, geringe Skalierung	gleichmäßig	-	fein	Person bewegt nur geringfügig den Kopf, Moiré-Effekt und Treppchenbildung im Testmuster erkennbar
sr016	38	Nikon S Tele 70- 210mm	42 - 58	geringe Translation, geringe Skalierung	Variation	-	fein	Person bewegt nur geringfügig den Kopf, Moiré-Effekt und Treppchenbildung im Testmuster erkennbar
sr017	10	Nikon S Tele 70- 210mm	42 - 47	geringe Translation	Variation	-	fein	Person bewegt nur geringfügig den Kopf
sr018	12	Nikon S Tele 70- 210mm	45 - 55	geringe Translation, geringe Skalierung	geringe Variation	-	fein	Person bewegt nur geringfügig den Kopf, Moiré-Effekt und Treppchenbildung im Testmuster erkennbar
sr019	3	Nikon S Tele 70- 210mm	45 - 55	geringe Translation	gleichmäßig	-	fein	Person bewegt nur geringfügig den Kopf, Moiré-Effekt und Treppchenbildung im Testmuster erkennbar
sr020	29	Nikon S Tele 70- 210mm	38 - 48	geringe Translation, geringe Skalierung	starke Variation	-	-	Person nähert sich der Kamera

Tabelle 8: Beschreibung der Testsets



Abbildung 36: Testset n004



Abbildung 37: Testset n006



Abbildung 38: Testset n013

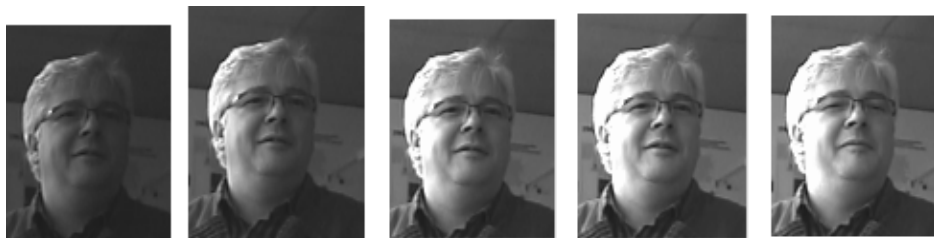


Abbildung 39: Testset b001



Abbildung 40: Testset b004



Abbildung 41: Testset b009



Abbildung 42: Testset r002



Abbildung 43: Testset r000



Abbildung 44: Testset r016

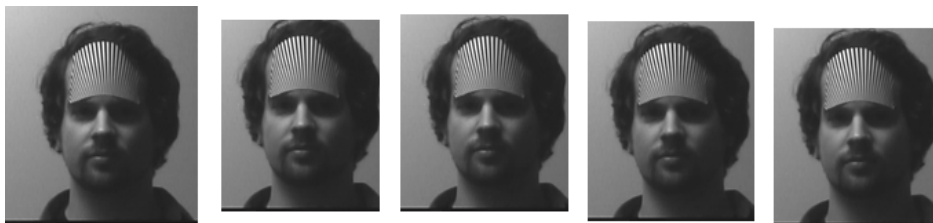


Abbildung 45: Testset sr019

DVD

Eidesstattliche Erklärung

„Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.“

8. Februar 2009, Münster