

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/267096327>

Automatische Generierung von Bildoperationsketten mittels genetischer Programmierung und CMA-Evolutionsstrategie

Conference Paper · November 2014

DOI: 10.13140/2.1.2403.5841

READS

120

3 authors:



[Roman Tobias Kalkreuth](#)

Technische Universität Dortmund

4 PUBLICATIONS 1 CITATION

SEE PROFILE



[Günter Rudolph](#)

Technische Universität Dortmund

155 PUBLICATIONS 4,335 CITATIONS

SEE PROFILE



[Jörg Krone](#)

Fachhochschule Südwestfalen

37 PUBLICATIONS 107 CITATIONS

SEE PROFILE

Automatische Generierung von Bildoperationsketten mittels genetischer Programmierung und CMA- Evolutionsstrategie

R. Kalkreuth¹, G. Rudolph¹, J.Krone²

¹Fakultät für Informatik,
Technische Universität Dortmund, Otto-Hahn-Str. 14, 44221 Dortmund
Tel. (0231) 755-7701 Fax (0231) 755-7740
E-Mail: {roman.kalkeuth, guenter.rudolph }@tu-dortmund.de

²Institut für Computer Science, Vision und Computational Intelligence,
Fachhochschule Südwestfalen, Frauenstuhlweg 31, 58644 Iserlohn
Tel. (02371) 566-303 Fax (02371) 566-209
E-Mail: krone.joerg@fh-swf.de

Abstract

Die in der digitalen Bildverarbeitung verwendeten Verfahren setzen sich häufig aus einer Aneinanderreihung von Bildoperationen (sog. Bildoperationsketten) zusammen. Möchte man genauere Informationen über die Operationen erhalten, welche auf ein Bild angewendet wurden, ist eine Rekonstruktion notwendig. Für eine schnelle und automatische Rekonstruktion empfiehlt sich der Einsatz eines Optimierungsverfahrens. Hierbei bietet sich die genetische Programmierung an, da eine Bildoperationskette als Chromosom im Sinne der genetischen Programmierung betrachtet werden kann. Durch den evolutionären Algorithmus kann durch die Rekonstruktion auch eine Optimierung erfolgen, in dem kürzere und damit weniger rechenintensive Wege gefunden werden. Neben der Rekonstruktion der Struktur einer Bildoperationskette ist auch das Auffinden der einzelnen Parameter von Bedeutung. Zum Auffinden der Parameter kann die *Kovarianz-Matrix-Adaptions* Evolutionsstrategie (CMA - ES) zum Einsatz kommen.

1 Einleitung

In der biomedizinischen Bildverarbeitung entsteht durch den wachsenden Einsatz von bildgebenden Verfahren eine hohe Dichte an Bildmaterial. Dieses Bildmaterial häufig im Rohzustand nicht direkt verwertbar und muss mithilfe verschiedener Verfahren aufgewertet werden.

Dies betrifft unter anderem die Verbesserung der Bildqualität sowie die Extraktion bestimmter Merkmale. Die Auswertung und Aufarbeitung des Bildmaterials wird häufig mithilfe von herkömmlichen Bildbearbeitungsprogrammen durchgeführt. Die hierbei verwendeten Verfahren setzen sich häufig durch eine Aneinanderreihung von Bildoperationen zusammen. Die sequenzielle Anordnung von Bildoperationen bezeichnet man auch als Bildoperationskette. In der biomedizinischen Bildverarbeitung werden häufig große Mengen an Bilddaten manuell über verschiedene Programme bearbeitet. Dies ist vor allem in der medizinischen Bildverarbeitung der Fall, wo viele Originalaufnahmen von Hand bearbeitet werden müssen, um entsprechende Ergebnisbilder zu erhalten. Für den Benutzer der Bildbearbeitungsprogramme ist die entstandene Kette mit ihren zugehörigen Parametern zwischen dem Original- und Ergebnisbild zu meist nicht genau einsehbar. Möchte man genauere Informationen über die Bildoperationskette erhalten, welche auf ein Bild angewendet wurde, ist eine Rekonstruktion der Kette notwendig. Die Rekonstruktion ermöglicht einen genauen Überblick, welche Bildoperationen und Parameter verwendet wurden, und kann zur Optimierung der Bildoperationskette dienen. Eine Optimierung umfasst das Auffinden von kürzeren und damit weniger rechenintensiven Ketten. Dies ist vor allem bei einer großen Menge an Bildmaterial hilfreich, da in der Summe Bearbeitungszeit für die Auswertung gespart werden kann. Über die Rekonstruktion kann beispielsweise die manuelle medizinische Auswertung von Bilddaten automatisiert werden, in dem man die gefundene Kette auf einen kompletten Bilddatensatz anwendet.

Die Rekonstruktion kann durch die Anwendung verschiedener Kombinationen aus einer festgelegten Auswahl an Bildoperationen geschehen. In den meisten Fällen ist eine manuelle Rekonstruktion sehr zeitaufwendig und ungenau. Auch das automatische Abarbeiten aller möglichen Kombinationen erfordert einen großen Rechenaufwand. Für eine genauere und schnellere Rekonstruktion empfiehlt sich der Einsatz eines Optimierungsverfahrens. Es konnte durch frühere wissenschaftliche Arbeit in [1] gezeigt werden, dass die genetische Programmierung eine gute Anpassungsmöglichkeit an die sich ergebene Problemstellung bietet. Eine Bildoperationskette kann als Chromosom im Sinne der genetischen Programmierung betrachtet werden. Durch eine gezielte Anpassung der genetischen Programmierung können kürzere Ketten gefunden werden. Das Verfahren, welches in [1] vorgestellt wurde, ist imstande eine Rekonstruktion sowie eine Optimierung der Struktur einer Bildoperationskette durchzuführen. Neben der Rekonstruktion der Struktur ist auch das Auffinden der einzelnen Parameter von Bedeutung. Viele Bildoperationen werden in der Bildverarbeitung mit unterschiedlichen Werten parametrisiert.

Die einzelnen Parameter können dabei in ihrem Wertebereich und Typ stark abweichen. Zum Auffinden der Parameter bietet die *Kovarianz-Matrix-Adaption Evolutionsstrategie* (CMA – ES) [2] als *State of the Art* Verfahren im Bereich der Evolutionsstrategien eine gute Kombinationsmöglichkeit mit der genetischen Programmierung. Die Parameter einzelner Operatoren können in einem Vektor zusammengefasst und der CMA-ES zur Optimierung übergeben werden. Die Kombination aus genetischer Programmierung und CMA-ES geschieht über einen evolutionären Algorithmus. Als Basis für die Entwicklung der Parameterrekonstruktion diene das Verfahren aus [1]. Der evolutionäre Algorithmus verbindet die Gebiete der evolutionären Optimierung und digitalen Bildverarbeitung. Man spricht hierbei auch von „evolutionärer Bildverarbeitung“.

2 Vorbetrachtung und Lösungskonzept

2.1 Analyse der Problemstellung

Die Problemstellung beinhaltet das Auffinden der Struktur der Bildoperationskette und dessen Parameter, welche vom Originalbild zum Zielbild führt (siehe Abb. 2.1). Das Zielbild ergibt sich häufig aus einer manuellen Bearbeitung des Original- bzw. Startbildes. Es sind keine genauen Informationen über die verwendeten Bildoperationen vorhanden. Meistens liefern die verwendeten Bildbearbeitungsprogramme nur grobe Informationen, welche Bildverarbeitungsverfahren oder Routinen hinter den einzelnen Funktionen des Programms stecken. Das Gebiet der verwendeten Bildoperationen kann daher lediglich eingegrenzt werden (z.B. Segmentierung oder Kantendetektion). Bevor die genetische Programmierung und die CMA-ES eingesetzt werden können, müssen im ersten Schritt Individuen definiert werden, welche die Problemstellung anhand der ihnen zugewiesenen Fähigkeiten bzw. Gene lösen können. Zur Einordnung der Ergebnisse der einzelnen Individuen muss zu dem eine Fitnessfunktion definiert werden, welche die Distanz zum Zielbild beschreibt. Damit sich nicht zu lange Ketten bilden, kann eine maximale Länge festgelegt werden. Für das Auffinden der Parameter können zudem Wertegrenzen festgelegt werden, um ergebnislose Operationen zu vermeiden.



Abb. 2.1. Problemstellung (links Startbild – rechts Zielbild)

2.2 Anpassung der Individuen

Für das Auffinden der Struktur muss ein Individuum definiert werden, welches eine Bildoperationskette repräsentiert und über eine Fitnessfunktion eingeordnet werden kann. Bei der genetischen Programmierung kann eine Bildoperationskette als Chromosom betrachtet werden, wobei jede einzelne Bildoperation ein Gen darstellt (siehe Abb. 2.2). Die Länge des Chromosoms kann von 1 bis zur maximalen Länge variieren. Jedes Individuum erhält damit eine Funktion, welche auf das Startbild angewendet werden kann. Das daraus resultierende persönliche Ergebnis kann zur Einordnung der Individuen verwendet werden. Für das Auffinden der Parameter muss das Individuum für den Einsatz der CMA-ES angepasst werden. Neben dem Chromosom müssen Vektoren für die einzelnen Parameter der Bildoperationen definiert werden. Für die CMA-ES sind zu dem noch Vektoren für Schrittweiten und Wertegrenzen notwendig (siehe Abb. 2.3). Die einzelnen Individuen können nach ihrer Anpassung in einer Population zusammengefasst werden.



Abb. 2.2 Chromosom aus drei Bildoperationen

$$\vec{X} = \begin{pmatrix} x1 \\ x2 \\ \dots \\ xn \end{pmatrix} \quad \vec{\sigma} = \begin{pmatrix} \sigma 1 \\ \sigma 2 \\ \dots \\ \sigma n \end{pmatrix} \quad \vec{l} = \begin{pmatrix} l1 \\ l2 \\ \dots \\ ln \end{pmatrix} \quad \vec{U} = \begin{pmatrix} u1 \\ u2 \\ \dots \\ un \end{pmatrix}$$

Abb. 2.3 Vektoren für Parameter, Schrittweiten und Wertegrenzen

2.3 Fitness und Selektion

Die Fitness der Individuen wird über die Distanz zwischen Ergebnisbild eines Individuums und dem Zielbild gebildet. Als genaues und simples Distanzmaß bietet sich die Summe der abweichenden Grauwerte an, welche *pixelweise* zwischen Bild A und Bild B gebildet wird (Manhattan-Distanz, siehe 2.3). Die Summe der abweichenden Grauwerte aus 2.3 kann als *Raw-Fitness* F_{raw} gewählt werden. Damit sich die Fitnesswerte zwischen 0 und 1 bewegen, wird aus der *Raw-Fitness* die *Adjusted-Fitness* F_{adj} nach [3] gebildet (2.4).

$$F_{raw} = \sum_{j=1}^{height} \sum_{i=1}^{width} |A[j, i] - B[j, i]| \quad (2.3)$$

$$F_{adj} = \frac{1}{(1 + F_{Raw})} \quad (2.4)$$

Ziel der Selektion ist es, gute Teilergebnisse für die Rekombination aufzufinden. Dies gilt zu gleich für die Strukturanpassung als auch für die Parameterentwicklung. Für die Selektion bietet sich zum einen die Turnirselektion als Selektionsstrategie an. Hierbei treten Individuen, welche zufällig aus der Population ausgewählt werden, in Turnieren gegeneinander an. Sieger eines Turniers ist das Individuum mit der höchsten Fitness. Der Vorteil dieser Selektionsstrategie liegt darin, dass sich Individuen mit guter Fitness durchsetzen. Der Selektionsdruck kann zu dem über die Turniergröße eingestellt werden. Es werden so viele Turniere durchgeführt, bis die erforderliche Anzahl an Eltern-Individuen für die Rekombination erreicht ist.

2.4 Genetische Anpassung der Chromosomen

Die genetische Anpassung der Chromosomen geschieht mittels Rekombination und Mutation im Sinne der genetischen Programmierung. Bei der Rekombination kann für kleinere Ketten (bis 10 Operationen) mit einem Kreuzungspunkt (*One-Point-Crossover*) gearbeitet werden. (siehe Abb. 2.4).

Von den zur Rekombination ausgewählten Eltern wird jeweils die Hälfte der Gene auf das neue Kind-Individuum übertragen. Für längere Ketten empfiehlt sich der Einsatz mehrerer Kreuzungspunkte (*Two-Point-Crossover*, *Uniform-Crossover*) um eine vielfältigere Mischung der Gene zu erreichen.

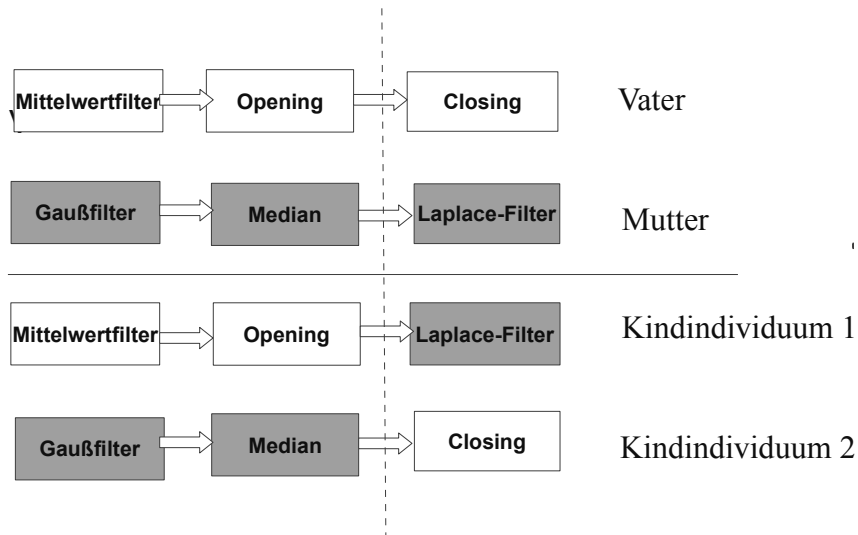


Abb. 2.4 Rekombination der Chromosomen mittels One-Point-Crossover

Enthalten die Eltern-Individuen gleiche Bildoperationen, so müssen deren Parameter und Schrittweiten rekombiniert werden. Dies wird für die Parameter \vec{X} und Schrittweiten $\vec{\sigma}$ mittels arithmetischem Crossover (2.7) durchgeführt. Hierbei wird die Rekombination von zwei Parametern P_1 und P_2 vorgenommen, wobei α zufällig gewählt wird.

$$P_{cross} = \alpha P_1 + (1 - \alpha) P_2 \quad \alpha \in [0, 1] \quad (2.7)$$

Neben der Rekombination wird auch eine Mutation der Chromosomen vorgenommen. In [1] wurde dies lediglich durch das Austauschen eines Gens durch ein zufälliges, neu gezogenes Gen durchgeführt. Um die Vielfältigkeit innerhalb der Population zu erhöhen, empfiehlt sich das Hinzufügen bzw. Entfernen, sowie das Vertauschen von Operationen als Mutation anzuwenden.

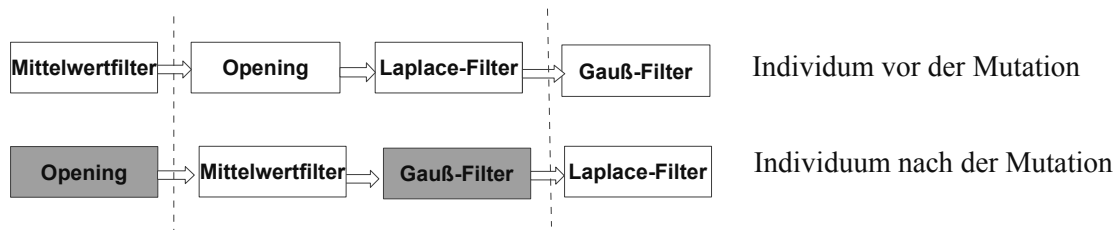


Abb. 2.5 Mutation durch Vertauschen mit 2 Vertauschungspunkten

2.5 Anpassung der Parameter

Nach der Strukturanpassung kann die Anpassung der Parameter erfolgen. Die Anpassung bzw. die Weiterentwicklung der Parameter für ein Individuum geschieht, wie schon erwähnt, über die CMA-ES. Gute Strukturergebnisse werden der CMA-ES zur Parameterentwicklung übergeben. Das heißt, dass nur Individuen mit entsprechender Fitness mittels CMA-ES weiterentwickelt werden. Es wird also nur eine gewisse Anzahl der besten Individuen aus der aktuellen Population verwendet. Die Struktur der Chromosomen bleibt bei der Anwendung der CMA-ES unverändert. Es werden lediglich die in Abb. 2.3. festgelegten Vektoren übergeben. Nach der Weiterentwicklung werden die Individuen wieder in die Population zurückgeführt. Durch den Einsatz von adaptiven Schrittweiten und „Suchrichtungen“ bzw. Lagewinkeln über die Kovarianzmatrix kann ein schneller Suchfortschritt beim Auffinden der Parameter erzielt werden. Bei der CMA-ES werden die in der Kovarianzmatrix enthaltenen Kovarianzen von Schrittweiten und Lagewinkeln in jeder Generation mit neuen Informationen aktualisiert, um diese nicht von Generation zu Generation neu berechnen zu müssen. Die Informationen basieren auf guten möglichen Lösungen, welche in der aktuellen Generation vorkommen. Ziel ist es, die Wahrscheinlichkeit von vormals guten Lösungsschritten zu erhöhen. Für das Update der Kovarianzmatrix stehen zwei Strategien zur Verfügung: das *Rank - μ Update* sorgt für eine gute Nutzung der Informationen für das Update aus einer Generation. Diese Update-Strategie empfiehlt sich bei großen Populationsgrößen. Das *Rank - one Update* hingegen nutzt Informationen für das Update generationsübergreifend und ist speziell wichtig bei kleineren Populationsgrößen. Abhängig von der Dimension eines Individuums bzw. seiner Parameter können *Rank- μ Update* oder *Rank-One Update* eingesetzt werden. Nach der Optimierung mittels CMA-ES kann die gewichtete Mittelwertlösung der besten Generation als beste Lösung verwendet werden. Diese wird letztlich in die aktuelle Population re-integriert.

2.5.1 Genetische Anpassung der Parameter

Die CMA-ES erzeugt in jeder Generation neue Lösungen über die Normalverteilung (2.9). Hierbei gibt g die aktuelle Generation an und $m^{(g)}$ den Mittelwert. Die globale Schrittweite $\sigma^{(g)}$ dient zur Skalierung der neuen Generation und $N(0, C^{(g)})$ „verzerrt“ die Normalverteilung gemäß der in der Kovarianzmatrix $C^{(g)}$ enthaltenen Kovarianzen von Schrittweiten und Lagewinkeln. Die genetische Anpassung wird für λ neue Lösungen durchgeführt

$$x_k^{(g+1)} = m^{(g)} + \sigma^{(g)} N(0, C^{(g)}) \text{ für } k = 1, \dots, \lambda \quad (2.9)$$

2.6 Evolutionärer Algorithmus

Im evolutionären Algorithmus werden die zuvor besprochenen Selektions- und Rekombinationsstrategien iterativ miteinander verbunden. Der evolutionäre Algorithmus wird hierzu in zwei Teile aufgeteilt. Der äußere Teil übernimmt die Entwicklung der Struktur mittels genetischer Programmierung. Der innere Teil des evolutionären Algorithmus übernimmt die Entwicklung der Parameter mittels CMA-ES. In jeder Generation des evolutionären Algorithmus wird nach der Strukturanpassung die Parameterentwicklung vorgenommen. Alternativ kann die Parameterentwicklung erst gestartet werden, wenn die Individuen einen gewissen Grad an Entwicklung durch die Strukturanpassung erreicht haben. Vor dem Start des Algorithmus wird eine Anfangspopulation erzeugt, welche die erste Generation bildet. Mit jeder Iteration wird über die Rekombination eine neue Generation gebildet. Vor jedem Durchlauf wird geprüft, ob das Abbruchkriterium erfüllt ist. Dies kann die optimale Fitness eines Individuums oder die maximale Anzahl an Generationen sein. Die optimale Fitness wäre bezogen auf die Problemstellung eine gewünschte Übereinstimmung mit dem Zielbild.

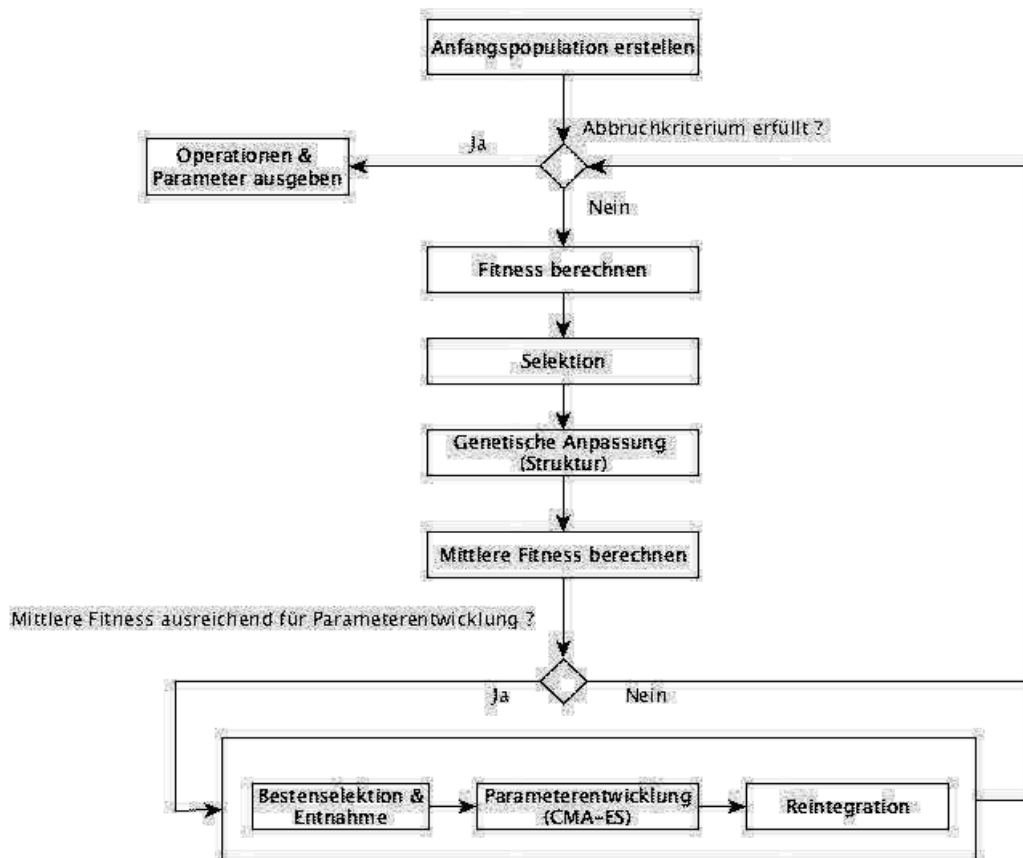


Abb. 2.6 Programmablaufplan evolutionärer Algorithmus

3 Praktische Umsetzung

Für die praktische Umsetzung des Lösungskonzepts wurde eine Java-Anwendung erstellt. Als Vorlage diente die Java-Anwendung aus [1].

3.1 Codierung und Implementierung der Bildoperationen

Die verschiedenen Bildoperationen werden mit Zahlen oder Buchstaben codiert. Auf diese Weise können die Bildoperationsketten bzw. Chromosomen als Permutationen dargestellt werden. Jede Bildoperation wird zu Beginn mit einer eindeutigen Zahl oder einem Buchstaben belegt.

Sobel-Operator	=>	4
Laplace-Filter	=>	7
Mittelwertfilter	=>	A
Gaussfilter	=>	9
Segmentierung	=>	5

Tabelle 3.1 Bildoperationen mit Bezeichner

Nach der Codierung werden entsprechende Permutationen gebildet. Die Länge der Permutationen variiert entsprechend von eins bis zur maximal festgelegten Kettenlänge. Für jene Bildoperationen, welche Parameter besitzen werden entsprechende Vektoren mit Zufallswerten gebildet.

Individuum 1 (Chromosom : 2691A)	$x_2 = (3, 10, 0.3)$ $x_A = (5)$
Individuum 2 (Chromosom : 560B)	$x_5 = (127)$ $x_B = (0.2)$
Individuum 3 (Chromosom : 4203F)	$x_0 = (3, 0.5)$ $x_F = (7, 0, 3)$

Tabelle 3.2 Drei gebildete Individuen mit Chromosomen und Parametervektoren

Die Bildoperationen wurden mit OpenCV [4] implementiert. OpenCV zählt zu den größten Bildverarbeitungsbibliotheken und bietet eine große Auswahl an Bildoperationen aus verschiedenen Bereichen der Bildverarbeitung.

3.2 CMA-Evolutionsstrategie

Für den Einsatz der CMA-ES wurde die Java Implementierung von [7] verwendet.

3.3 Evolutionärer Algorithmus

Der evolutionäre Algorithmus wurde nach dem Programmablaufplan aus Abb. 2.6 umgesetzt. Der Algorithmus wurde in zwei Funktionen unterteilt, welche jeweils die Struktur- bzw. die Parameterentwicklung vornehmen. Notwendige Parameter wie die Anzahl der Generationen oder die Anzahl von Eltern- und Kindindividuen werden vor dem Start des Algorithmus separat initialisiert.

Das Abbruchkriterium stellt, wie schon beschrieben, eine Restdistanz zum Zielbild oder eine festgelegte maximale Anzahl an Generationen dar. Neben dem Ablauf aus Selektion, Rekombination und Mutation wird in jeder Generation die durchschnittliche Fitness der aktuellen Generation berechnet und geprüft, ob diese für die Parameterentwicklung ausreichend ist. Für die Parameterentwicklung werden n Individuen aus der Elite entnommen, welche dann mittels CMA-ES weiterentwickelt werden. Die Selektion aus der Elite kann nach zwei Kriterien geschehen : Selektion der besten Individuen oder per Zufall. Den mittels CMA-ES entwickelten Individuen wird die Mittelpunktlösung der CMA-ES zugewiesen bevor diese wieder in die aktuelle Generation re-integriert werden.

Evolutionärer Algorithmus (Abbruchkriterium, CMA-ES Startfitness, Genetische Parameter)

```

1   Elternpopulation ← erzeuge Anfangspopulation
2   while Abbruchkriterium nicht erfüllt
3   do berechne Fitness von Elternpopulation
4     Eltern ← Selektion(Elternpopulation)
5     Kindpopulation ← Rekombination(Eltern)
6     Mutaton(Kindpopulation)
7     Mittlere_Fitness← berechne mittlere Fitness (Kindpopulation)
8     if Mittlere_Fitness >= CMA-ES Startfitness
9     then Elite ← Bestenselektion(Kindpopulation)
10      CMA_Individuen ← Entnehme n Individuen aus Elite
11      Parameterentwicklung(CMA_Individuen, Genetische Parameter)
12      Integriere CMA_Individuen in Kindpopulation
13   Best ← Selektiere bestes Individuum
14   return Best

```

Listing 3.1 Evolutionärer Algorithmus

Parameterentwicklung (CMA_Individuen, Genetische Parameter)

```

1   for each Individuum in CMA_Individuen
2   do CMA-Lösungen ← CMA-ES (Individuum)
3     Individuum ← Mittelpunktbildung(CMA-Lösungen)
4   return Elite

```

Listing 3.2 Parameterentwicklung

4 Ergebnisse

Um das Verfahren zu testen, wurde der Benchmark Nr. 6 aus der biomedizinischen *Broad Bio Image Benchmark Collection* [5] verwendet. Der Benchmark enthält Problemstellungen im Bereich der Segmentierung und Zellerkennung, welche sich jeweils aus Originalaufnahmen und Referenzbildern (*Ground-Truth*) zusammensetzen.

Die kontrastarmen Aufnahmen enthalten menschliche U2OS-Zellen. Die Referenzbilder wurden vom *Broad-Institut* segmentiert und die Anzahl der Zellen festgehalten. Der Benchmark eignet sich sehr gut für eine praxisnahe Anwendung der evolutionären Bildverarbeitung. Über den evolutionären Algorithmus soll die Bildoperationskette zwischen den Rohbildern und den Referenzbildern (Abb. 4.1) rekonstruiert werden.

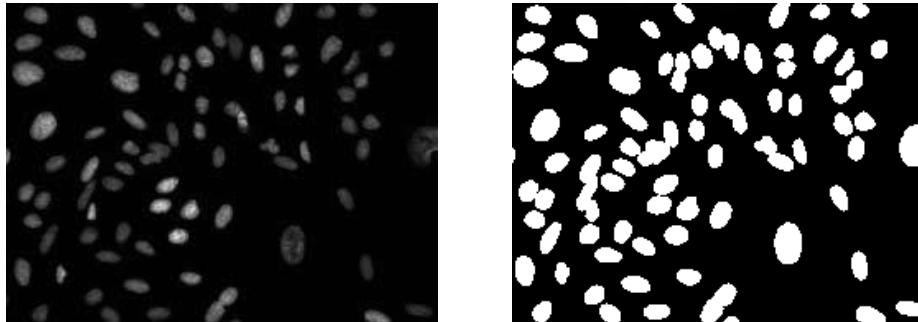


Abb. 4.1 Kontrastoptimierte Originalaufnahme (links) und Ground-Truth(rechts) aus dem Benchmark Nr. 6

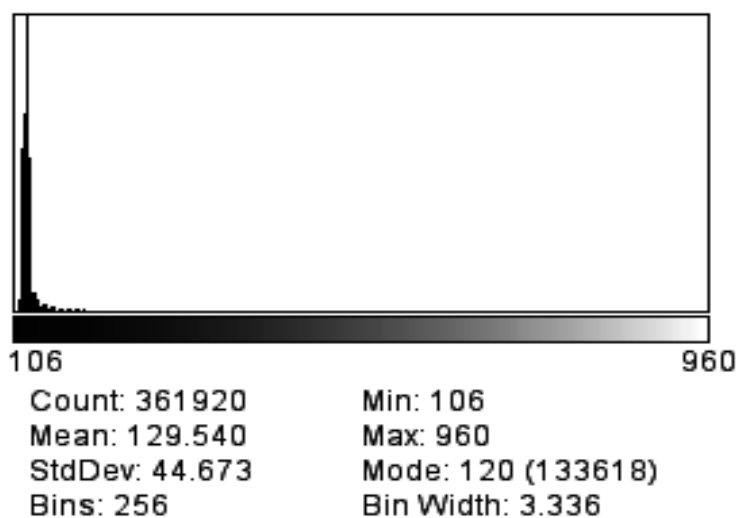


Abb. 4.2 Histogramm einer Originalaufnahme des Benchmarks

4.1 Auswertung des Benchmark

Aus dem Datensatz Nr. 16 (Aufnahmen im Fokus) mit 384 Zellbildern wurde eine Stichprobe von 32 Bildern entnommen. Der Benchmark wurde unter dem Gesichtspunkt der Segmentierung ausgewertet und die Referenzbilder entsprechend binarisiert. Dem evolutionären Verfahren lagen zur Rekonstruktion die Originalaufnahme und das Ground-Truth vor.

Um die Ergebnisse einordnen zu können, wurde als Vergleichsverfahren das sogenannte Otsu-Verfahren[8] ausgewählt. Das Otsu-Verfahren ist ein automatisches Segmentierungsverfahren, welches mit statistischen Mitteln arbeitet. Um die Anzahl der Zellen in den Ergebnisbildern beider Verfahren sowie der binarisierten Referenzbilder zu ermitteln, wurde die Objekterkennung des *Cell-Profilers*[6] verwendet. Für den Vergleich wurde die relative Abweichung F (4.1) zwischen Anzahl der Zellen im Ergebnisbild der beiden Verfahren n und der Anzahl der Zellen im Referenzbild n_{ref} gebildet. Neben der Anzahl der Zellen wurde auch die mittlere Größe des Zellbereiches (*mean cell area*) für die Auswertung der Ergebnisse einbezogen.

$$F = \frac{|n - n_{ref}|}{n_{ref}} \quad (4.1)$$

Bei der Segmentierung können durch einen falschen Schwellwert, kleine Objekte entstehen, welche fälschlicherweise als Zellen erkannt werden. Um diese Objekte zu filtern, wurde auf das Ergebnisbild des Otsu-Verfahrens eine Opening-Operation jeweils mit 3x3 und 5x5 ($S=3, S=5$) angewendet. Der Suchraum des evolutionären Ansatzes setzte sich aus folgenden Operationen zusammen : globale und lokale Segmentierungsverfahren, Rauschunterdrückung, morphologische Operationen und Watershed-Verfahren. Tabelle 4.1 zeigt die wichtigsten Parameter, mit welchen der evolutionäre Algorithmus eingestellt wurde. Für die Problemstellung waren maximal 80 Individuen ausreichend. Die Anzahl der Individuen, welche mittels CMA-ES weiterentwickelt wurden (*CMA-Individuen*), lag bei 5. Bei der Bearbeitung des Benchmarks variierte die Länge der Chromosomen zwischen 2 und 6 Operationen. Die Dimension Parameter lag abhängig vom Testfall zwischen 4 und 11 Parametern.

Genetische Programmierung

Größe Elternpopulation : 60

Größe Kindpopulation : 80

Anzahl Elite : 10

Max. Anzahl Generationen : 10

Mutationswahrscheinlichkeit : 0.2

Selektionsstrategie : Turnier-Selektion

CMA-ES

$\mu=10$, $\lambda=20$

Max. Durchläufe = 20

Anzahl CMA-Individuen : 5

Tabelle 4.1 Wichtige Parameter für den evolutionären Algorithmus

Images	GT		EA		Otsu S=3		Otsu S=5	
	n_{ref}	C_{mean}	$F(\%)$	C_{mean}	$F(\%)$	C_{mean}	$F(\%)$	C_{mean}
a01_s2	11	1078	18,0	838	100,0	0	100,0	0
a07_s2	40	1580	5,0	1499	27,5	727	25,0	723
b01_s1	44	1393	2,2	1405	27,3	659	27,3	653
b13_s1	85	1403	4,7	1388	21,1	990	17,6	1010
c02_s1	82	1484	0,0	1479	19,5	1121	19,5	1113
c11_s1	67	1749	5,9	1646	35,8	1173	35,8	1162
d03_s2	86	1498	0,0	1486	22,1	1096	22,1	1087
d18_s1	64	1591	4,6	1516	12,5	1124	10,9	1130
e06_s1	70	1337	2,8	1326	10,0	1061	7,1	1075
e21_s2	76	1642	6,5	1792	34,2	1068	30,2	1092
f01_s1	63	1606	1,5	1544	14,2	1235	14,2	1223
f15_s1	93	1647	1,1	1679	28,0	1117	28,0	1117
g04_s2	79	1545	3,7	1598	12,7	1249	14,1	1208
g23_s1	87	1417	5,7	1499	13,8	1087	14,9	1070
h01_s1	40	1484	0,0	1499	17,5	762	17,5	752
h19_s2	79	1023	10,6	1300	15,2	1088	11,4	1094
i02_s1	85	1421	7,1	1322	9,4	1170	8,2	1170
i11_s2	80	1390	7,5	1300	15,0	1051	15,0	1045
j07_s1	83	1387	6,0	1320	22,9	1012	18,0	1034
j18_s2	76	1501	3,9	1443	22,4	1042	19,7	1051
k09_s1	64	1328	0,0	1330	12,5	1051	12,5	1044
k12_s1	77	1496	3,8	1466	16,9	1115	15,6	1118
l03_s2	77	1686	7,7	1835	26,0	1197	24,7	1200
l20_s2	84	1797	0,0	1799	40,5	1101	44,0	1061
m08_s1	66	1423	1,5	1442	14,9	1064	18,2	1022
m24_s2	67	1366	5,9	1288	13,4	1049	13,4	1039
n10_s2	80	1466	5,0	1388	17,5	1037	13,9	1031
n14_s1	68	2053	2,9	2017	35,3	1463	36,8	1438
o05_s1	83	1390	1,2	1387	18,1	1023	15,7	1033
o19_s2	83	1523	2,4	1475	18,1	1103	18,1	1093
p02_s1	51	1258	9,8	1356	11,8	620	11,8	612
p13_s1	60	1335	6,6	1280	6,6	1059	6,6	659
μ	70	1495	4,8	1467	22,3	1022	21,5	1005

GT Ground Truth
EA Evolutionärer Ansatz
F(%) Relative Abweichung in %
Cmean Mean cell area

Tabelle 4.2 Ergebnisse Benchmark BBBC 6

Die Auswertung der Ergebnisse in Tabelle 4.3 zeigt, dass der evolutionäre Ansatz die Segmentierungsprobleme besser lösen kann, als das Otsu-Verfahren. Die prozentuale Abweichung $F(\%)$ fällt mit 4.8% geringer aus, als das Otsu-Verfahren mit $> 20\%$. Auch der mittlere Zellbereich liegt näher an der Referenz als das Otsu-Verfahren. Der Vorteil des Otsu-Verfahrens liegt darin, dass keine Referenz zur Lösung eines Segmentierungsproblems benötigt wird. Der evolutionäre Ansatz benötigt zur Lösung zwar ein Referenzbild, ist aber im Stande alternative Ketten mit ihren entsprechenden Parametern zu finden. Abb. 4.1 zeigt den Vergleich eines Einzelergebnis zwischen Ground-Truth, evolutionärem Ansatz und Otsu-Verfahren. Man kann erkennen, dass mithilfe des evolutionären Ansatzes mehr Objekte segmentiert werden können, als mit dem Otsu-Verfahren. Allerdings weist auch das Ergebnisbild es evolutionären Ansatzes noch falsche Objekte auf. Für eine sehr genaue Zellerkennung wie sie im Ground Truth zu sehen ist, müssen noch komplexere Zellerkennungsalgorithmen im evolutionären Ansatz implementiert werden.

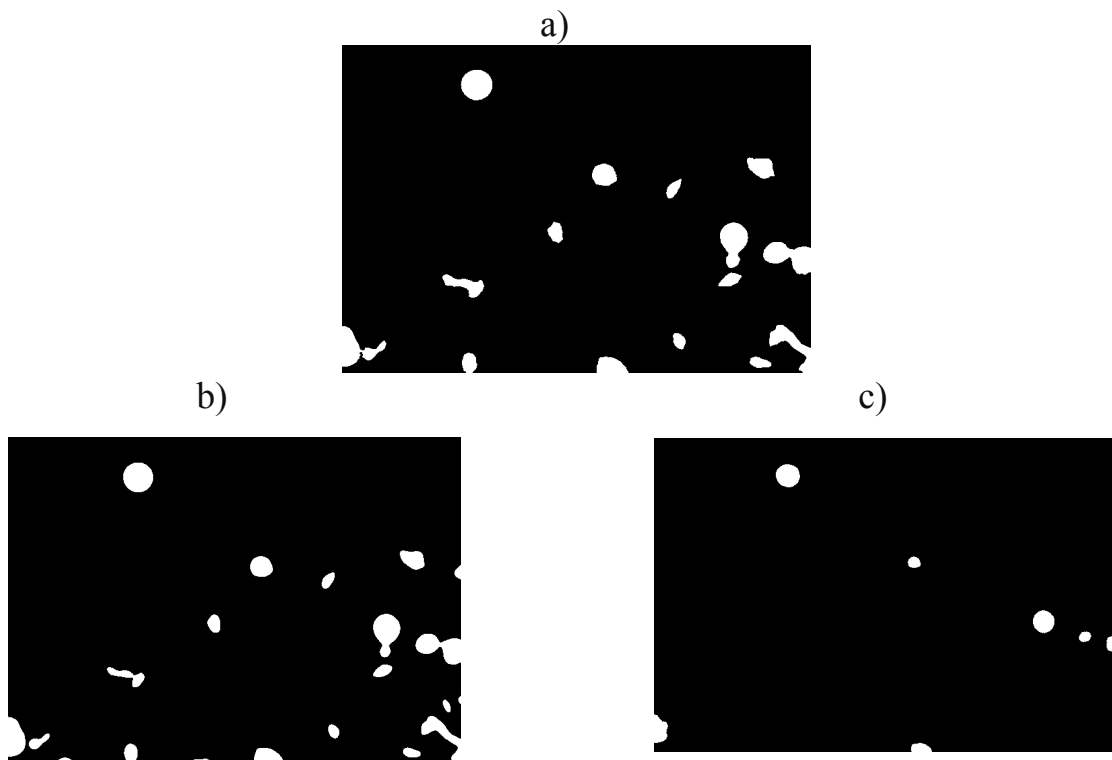


Abb. 4.1 Vergleich eines Einzelergebnis : a) Ground-Truth b) Ergebnis des evolutionären Ansatzes c) Ergebnis des Otsu-Verfahrens

5 Zusammenfassung und Ausblick

Es konnte gezeigt werden, dass mittels evolutionärer Optimierung reale Bildverarbeitungsprobleme aus der biomedizinischen Bildverarbeitung gelöst werden können. Die Kombination aus genetischer Programmierung und Evolutionsstrategie ist imstande, Struktur und Parameter einer Bildoperationskette zu rekonstruieren und zu optimieren. Wie schon beschrieben, werden in der biomedizinischen Bildverarbeitung große Mengen an Bilddaten von Hand ausgewertet. Der evolutionäre Algorithmus kann über die genaue Rekonstruktion der eingesetzten Bildoperationen zur Ableitung eines automatischen Verfahrens dienen. Als effektiv stellte sich beim Einsatz der CMA-ES die Re-Integration der Mittelpunktlösung heraus. Die Re-Integration der Mittelpunktlösung ist der Lösung des besten Individuums vorzuziehen, da diese einen größeren Lösungsraum repräsentiert als die beste Punktlösung. Ein Nachteil des Verfahrens stellt der hohe Rechenaufwand dar. Durch die Schachtelung der CMA-ES innerhalb der genetischen Programmierung müssen viele Fitnessberechnungen mittels Bildvergleich durchgeführt werden. Für ein schnelleres und effizienteres Auffinden einer Bildoperationskette muss die Fitnessberechnung bzw. das Distanzmaß optimiert werden. Hierbei kann zum einen die Berechnung der Fitnessfunktion parallelisiert werden oder der evolutionäre Algorithmus auf Teilalgorithmen aufgespalten werden. Des Weiteren sollte der Einsatz von effizienten Verfahren zum Vergleichen von Bildern untersucht werden. Hierbei bietet sich u.a. der Einsatz von sogenannten *Keypoint-Verfahren* an, welche den Vergleich auf signifikante Merkmalsbereiche beschränken. Das Verfahren arbeitet derzeit nur mit linearer genetischer Programmierung. Der Einsatz von Verzweigungen bzw. Baumstrukturen stellt eine wichtige Erweiterung für die Rekonstruktion und Optimierung von komplexeren Bildverarbeitungsverfahren dar. In Zukunft sollen auf Basis von Baumbasierter genetischer Programmierung weitere Benchmarks mit industriellem und biomedizinischen Ursprung bearbeitet werden.

Literatur- und Quellangaben

[1] Kalkreuth, Roman; Krone, Jörg; Schneider, Michael (2012): „*Automatische Generierung von Bildoperationsketten mittels genetischer Programmierung*,“ Fachhochschule Südwestfalen, Institut für Computer Vision & Computational Intelligence, Erschienen in: *Proceedings 22. Workshop Computational Intelligence*, Seite(n) 325–340, 2012, ISBN 978-3-86644917-6.

[2] Hansen, Nikolaus (1998) : „*Verallgemeinerte individuelle Schrittweisenregelung in der Evolutionsstrategie : Eine Untersuchung zur entstochastisierten, koordinatensystemunabhängigen Adaptation der Mutationsverteilung*“, Dissertation, Mensch & Buch Verlag, Berlin, ISBN 978-3933346292.

[3] Koza, J.R. (1992) : „*Genetic Programming : On the Programming of Computers by Means of Natural Selection*“, The MIT Press, ISBN 978-0262111706.

[4] OpenCV : *Open Source Computer Vision*, URL : www.opencv.org
Abrufdatum : 10.08.2013

[5] BBBC : *Broad Bioimage Benchmark Collection*, Broad Institute, URL : www.broadinstitute.org/bbbc/
Abrufdatum : 10.08.2013

[6] CellProfiler : *cell image analysis software*, Broad Institute, URL : www.cellprofiler.org
Abrufdatum : 10.08.2013

[7] CMA-ES Source Code, URL : www.lri.fr/~hansen/cmaes_inmatlab.html
Abrufdatum : 10.08.2013

[8] Otsu, Nobuyuki (1979) : „*A threshold selection method from gray-level histograms*“, Erschienen in : „*Systems, Man and Cybernetics*, IEEE Transactions on Volume:9; Issue: 1“, Seite(n) : 62–66