

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/266398778>

# Automatische Generierung von Bildoperationsketten mittels genetischer Programmierung

Conference Paper · December 2012

DOI: 10.13140/2.1.2753.9209

---

READS

62

3 authors, including:



[Roman Tobias Kalkreuth](#)

Technische Universität Dortmund

4 PUBLICATIONS 1 CITATION

SEE PROFILE



[Jörg Krone](#)

Fachhochschule Südwestfalen

37 PUBLICATIONS 107 CITATIONS

SEE PROFILE

# **Automatische Generierung von Bildoperationsketten mittels genetischer Programmierung**

**R. Kalkreuth, J. Krone, M. Schneider**

Institut für Computer Science, Vision und Computational Intelligence,  
Fachhochschule Südwestfalen, Frauenstuhlweg 31, 58644 Iserlohn  
Tel. (02371) 566-303 Fax (02371) 566-209  
E-Mail: { Krone, MSchneider }@fh-swf.de  
Roman.T.Kalkreuth@stud.fh-swf.de

## **1 Einleitung**

Durch den wachsenden Einsatz von bildgebenden Verfahren in medizinischen und industriellen Bereichen ist der Einsatz der digitalen Bildverarbeitung nicht mehr wegzudenken. Die daraus entstehende hohe Dichte an Bildmaterial muss oft aufgearbeitet werden. Dies betrifft vorwiegend die Verbesserung der Bildqualität sowie die Extraktion bestimmter Informationen, welche im Rohzustand nicht erkennbar sind. Der Einsatz geschieht beispielsweise bei medizinischen Diagnoseverfahren ( z.B. CT, Röntgen oder MRT) oder bei industriellen Bildverfahren ( z.B. automatische Endkontrolle). Die Verarbeitung der Bilder kann während des jeweiligen Prozesses oder im Nachhinein geschehen. Dies geschieht über sogenannte Bildoperationen. Diese dienen zur Bildverbesserung sowie der Erkennung von Umrisen und Strukturen von Objekten. Die Erkennung von Strukturen wird auch als Kantendetektion bezeichnet. Dieses Verfahren ist für eine Objekterkennung wie sie oft gebraucht wird, in vielen Fällen unabdingbar. Meistens müssen Bilder auf welche eine Kantendetektion angewendet wird, durch vorgehende Operationen aufbereitet werden. Hierbei entstehen Anordnungen von Operationen welche eine Bildoperationskette definieren. In den meisten Fällen entstehen solche Ketten durch die Anwendung verschiedener Operationen in herkömmlichen Bildbearbeitungsprogrammen. Für den Benutzer ist die entstandene Kette zu meist nicht genau einsehbar und kann nicht auf eine größere Ansammlung von Bildern angewendet werden. Ein Verfahren welches die Kette an Operationen rekonstruieren kann wäre für Anwendungen in der Medizin und Industrie von großem Nutzen. Zu Anfang muss dabei eine vorher festgelegte Auswahl an Operationen vorhanden sein. Diese müssen so ausgewählt werden, dass sie das Anwendungsgebiet des zu bearbeitenden Bildes am besten treffen.

Durch die Anwendung verschiedener Operationen auf das Startbild entsteht ein hoher Rechenaufwand der sich selbst bei sehr schnell arbeitenden Systemen bemerkbar macht. Dies würde den zeitlichen Rahmen der Laufzeit vor allem bei längeren Ketten überschreiten. Man kann sich in Hinblick auf die Problemstellung einem Optimierungsverfahren bedienen, welches in der Programmierung sehr vielseitig einsetzbar ist. Die genetische Programmierung (GP) eignet sich in der Bildverarbeitung als sehr gut durchführbares Verfahren zur Optimierung der gegebenen Problemstellung. Die Informationen mit welcher die GP arbeitet können ohne größeren Aufwand aus dem bildverarbeitenden Teil der Programmierung extrahiert und übertragen werden. Die GP arbeitet mit einem evolutionären Algorithmus (EA) welcher die Grundprinzipien der biologischen Evolution zur Optimierung anwendet. Die Kombination aus Bildverarbeitung und EA wird unter dem Begriff „evolutionäre Bildverarbeitung“ (EB) zusammengefasst.

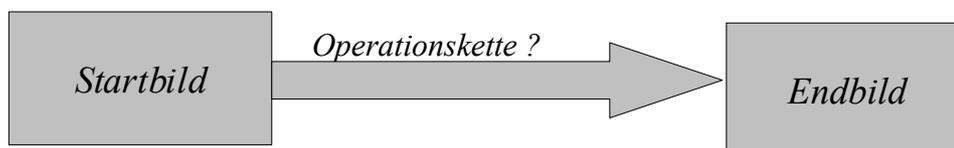


Abb. 1.1 Anwendung „evolutionäre Bildverarbeitung“

## 2 Vorbetrachtung

Vor der eigentlichen Umsetzung wurde über eine Vorbetrachtung die Problemstellung genauer analysiert und Vorbereitungen getroffen. Dies betraf vorwiegend die Anpassung der genetischen Programmierung auf die Problemstellung.

### 2.1 Analyse der Problemstellung

Die Problemstellung besteht im Auffinden einer Kombination aus Filtern, welche zwischen einem Start und Ergebnisbild stehen. Die Problemstellung gibt keinen Aufschluss darüber welche Filter eingesetzt wurden beziehungsweise aus welchem Bildverarbeitungsbereich diese stammen. Die Filter können lediglich eine entsprechende Auswahl an häufig verwendeten Filtern der industriellen sowie medizinischen Bildverarbeitung darstellen. Die Bilder werden durch die Anwendung der Filter in ihrer Breite und Höhe nicht verändert, so dass ein Bildvergleich zwischen Start- und Endbild problemlos erfolgen kann.

Die maximale Länge der Filterkette kann anfangs bei fünf Filtern festgelegt werden, eine spätere Anhebung auf sieben Filter kann zusätzlich in Betracht gezogen werden. Um die GP als Optimierungsverfahren einzusetzen, müssen die Gegebenheiten der Problemstellung entsprechend auf die GP angepasst werden. Dies betrifft in erster Linie die Anpassung der Individuen und dessen genetische Anpassungen welche im Sinne der GP durchgeführt werden. Ein weiterer wichtiger Aspekt bei der Nutzung der GP ist die Bewertung eines Individuums. Jedem Individuum werden Fähigkeiten zugewiesen welche es eindeutig bestimmt. Bezogen auf die Lösung der Problemstellung entwickelt jedes Individuum eine gewisse Stärke welche auch als Fitness bezeichnet wird. Für die Ermittlung der Fitness eines Individuums muss eine entsprechende Fitnessfunktion auf die Problemstellung angepasst werden.

## 2.2 Anpassung von Genen und Individuen

Die Anpassung verschiedener Individuen ist für den Einsatz der genetischen Programmierung ein hinreichendes Kriterium. Die Informationen, die man aus einer Filterkette gewinnen kann, müssen entsprechend auf ein Individuum abgebildet werden. Ein Filter kann dabei ein Gen repräsentieren Die genaue Reihenfolge der Filter, welche die Kette beinhaltet, ist ein Merkmal welche die Eindeutigkeit ermöglicht. Somit kann die Reihenfolge einer Kette auf das Chromosom eines Individuums übertragen werden. Jedem Individuum kann damit eine Funktion zugewiesen werden in welcher es die Reihenfolge entsprechend seiner Gene abarbeitet. Jedem Individuum steht das Startbild der Problemstellung zur Verfügung. Aus seiner Funktion ergibt sich dann entsprechendes Ergebnisbild mit dessen Hilfe die Fitness ermittelt werden werden kann. Die Größe eines Individuums kann dabei von eins bis hin zur maximal festgelegten Größe variieren. Ein mehrfaches Vorkommen eines Filters in den Genen ist möglich. Dies sollte aber als Option gewählt werden, da eine mehrfache Anwendung bestimmter Filter das Bild zu dunkel werden lässt und damit zum Verlust von Information beiträgt. Durch die Erzeugung vieler Individuen kann die Ansammlung in einer Population zusammengefasst werden.

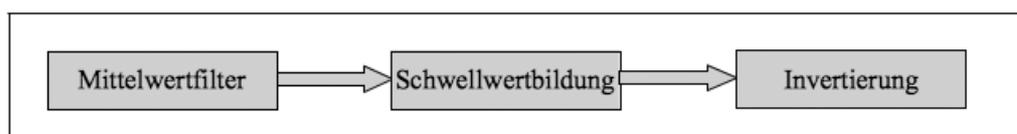


Abb. 2.1 Chromosom aus drei Filtern

## 2.3 Genetische Anpassung

Um Teilergebnisse weiterzuentwickeln und fortzuführen sind genetische Anpassungen notwendig. Den erzeugten Individuen wird das Recht der Fortpflanzung „zugesprochen“, damit sie ihre Gene weitergeben können. Die Rekombination bestimmter Individuen erzeugt damit die passende Folgegeneration in welcher die passenden Gene für die Problemstellung enthalten sind.

Im Vorfeld müssen zwei Elternteile vorhanden sein. Für die Rekombination zweier Individuen bietet sich ein sogenannter *One-Point-Crossover* an. Bei diesem Crossover wird mit Hilfe eines Kreuzungspunktes die Hälfte der Vatergene sowie die Hälfte der Muttergene auf ein neues Individuum übertragen. Die Anzahl der Elterngene bildet dabei die Anzahl der Kindgene. Abb. 2.2 zeigt die Bildungsvorschrift um die Genlänge eines Kindes mithilfe der Vater- und Muttergene zu bestimmen.

$$Kind_{Gene} = \begin{cases} Mutter_{Gene} & , \text{ falls } Mutter_{Gene} = Vater_{Gene} \\ \left\lceil \frac{(Vater_{Gene} + Mutter_{Gene})}{2} \right\rceil & , \text{ falls } Mutter_{Gene} \neq Vater_{Gene} \end{cases}$$

Abb. 2.2 Bildungsvorschrift Genlänge Kind

Die Anwendung eines *One-Point-Crossover* auf zwei Elternteile lässt zwei Kinder (sog. Genome) entstehen. Da aber nur ein Kind aus der Rekombination entstehen soll wird aus den beiden Genomen per Zufall das Kind bestimmt. Abb. 2.3 zeigt die Anwendung des Crossover auf die Chromosomen von Vater und Mutter. Die beiden Genome erben sich dabei entsprechend aus den Hälften der jeweiligen Elterngene. Die Genome unterscheiden sich in der Kombination der beiden Hälften. Bei Genom 1 wurde die vordere Hälfte vom Vater und die hintere Hälfte der Mutter gewählt. Bei Genom 2 entsprechend in anderer Reihenfolge.

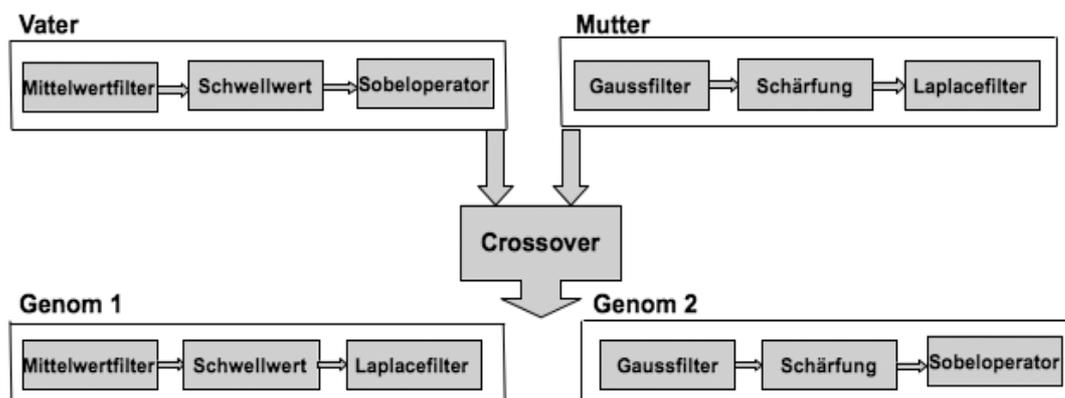


Abb. 2.3 Rekombination

Neben der Rekombination geschieht durch die Mutation eine genetische Anpassung. Mithilfe der Mutation kann die Vielfalt innerhalb der Population erhalten und erhöht werden. Die Mutation geschieht durch das Auswechseln eines Filters innerhalb der Kette gegen ein zufällig gewähltes Filter. Die Mutation wird nach der Rekombination angewandt. Die Mutation wird nicht auf die gesamte Population angewandt. Mithilfe einer Zufallsvariable wird eine gewisse Anzahl an Kindindividuen nach der Rekombination durch die Mutation verändert.

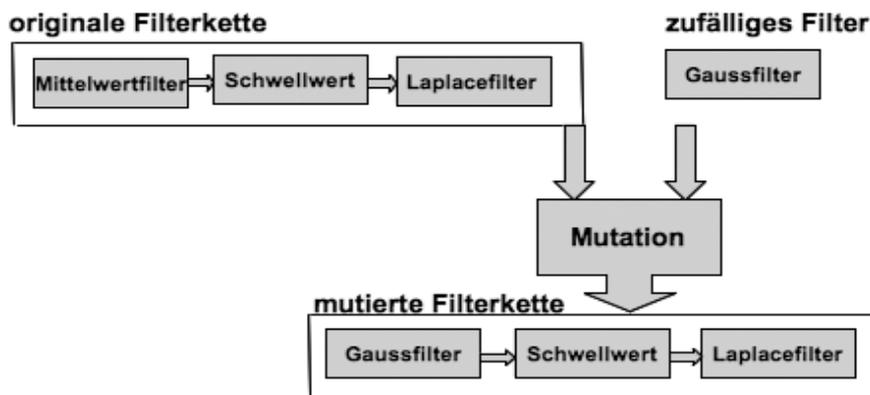


Abb. 2.4 Mutation

## 2.4 Bewertung und Selektion

Die Fitness beschreibt das Maß an Anpassung eines Individuums welches es in Bezug auf die Problemstellung entwickelt. Durch die Anwendung der Fähigkeiten des Individuums auf die Problemstellung entsteht ein Ergebnis. Das Ergebnis eines einzelnen Individuums wird unter den vorher festgelegten Kriterien welche das gewünschte Endergebnis erfüllen soll bewertet. Die Fitness kann beispielsweise die Distanz zwischen gewünschtem Endergebnis und dem aktuellen Ergebnis des Individuums sein. Die Distanz welche sich zwischen Ergebnisbild und dem Endbild ergibt kann als Fitness gewählt werden. Die einfachste Bewertungsmethode für einen Bildvergleich ist die sogenannte Pixeldistanz. Hierbei werden beide Bilder durchlaufen und die jeweils die beiden aktuellen Pixel mit einander verglichen. Gibt es eine Abweichung so wird der Betrag der beiden Pixel aufsummiert. Die Summe wird daraufhin durch die Anzahl der Pixel geteilt um das Verhältnis zwischen Abweichungssumme und Bildgröße zu ermitteln. Durch diese Berechnung erhält man die prozentuelle Abweichung. Eine Normalisierung erfolgt mit  $\frac{1}{256}$ , wobei 256 die Anzahl der Grauwerte repräsentiert. Durch die Normalisierung erhält man Werte welche sich zwischen 0.0 und 1.0 bewegen. Diese werden letztlich als Fitness gewählt.

$$D_{Summe} = \sum_{j=1}^{height} \sum_{i=1}^{width} |(A[j, i] - B[j, i])|$$

$$D_{Prozent} = \frac{D_{Summe}}{(width * height)}$$

$$D_{Norm} = \frac{D_{Prozent}}{256}$$

Abb. 2.5 Bildungsvorschrift Pixeldistanz

Neben der Bewertung der Fähigkeiten eines Individuums ist die Auswahl von Individuen aus einer Population von großer Wichtigkeit. Die Selektion ist eine Methode der GP zur Ermittlung und Auswahl der Individuen welche für die Rekombination am besten geeignet sind. Um möglichst gute Eltern für die Rekombination zu finden müssen bei der Selektion gut angepasste Individuen gefunden werden. Diese werden wie schon beschrieben durch ihre Fitness repräsentiert. Das geeignetste Verfahren hierzu stellt die sogenannte Turnierselektion dar. Diese lässt Individuen in Turnierrunden gegeneinander antreten und ermittelt so das beste Individuum aus der Auswahl. Über dieses Verfahren kann eine gezielte Fortführung entsprechender Gene erfolgen. Turniergröße, sowie die Anzahl der Runden können entsprechend den genetischen Parametern (z.B. Populationsgröße) angepasst werden.

## 2.5 Evolutionärer Algorithmus

Die GP arbeitet wie schon beschrieben mit Individuen welche in einer Population zusammengefasst sind. Eine genetische Anpassung im Rahmen von Rekombination und Mutation mit nur einem Durchlauf würde die Problemstellung nicht lösen können. Eine iterative Vorgehensweise ist für viele Problemstellungen daher unabdingbar. Der EA wendet innerhalb einer vorher festgelegten Anzahl an Iterationen die entsprechenden Anpassungen auf eine Population an. 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. Der Algorithmus muss auf die Problemstellung bezogen nicht besonders angepasst werden. Der normale Ablauf bestehend aus Bewertung – Selektion – Anpassung kann übernommen werden.

### 3 Praktische Umsetzung

Die praktische Umsetzung erfolgte mit einem Javaprogramm. Alle erforderlichen Mittel der Bildverarbeitung und GP wurden dazu ohne Rückgriff auf Frameworks implementiert. Das Programm bildete die Basis für die spätere Bearbeitung von Testfällen. Die wichtigsten Teile der praktischen Umsetzung sollen in diesem Kapitel beschrieben werden.

#### 3.1 Bildoperationen

Die Implementierung von Bildoperationen umfasste Faltungsoperationen, Punktoperationen und morphologische Operationen. Mithilfe der Faltung konnten wichtige Operationen zur Bildverbesserung und Kantendetektion umgesetzt werden. Als wichtigste Punktoperation wurde die Segmentierung umgesetzt. Diese ist bei der Anwendung von Kantendetektoren sehr wichtig und wird bei der Verarbeitung von Grauwertbildern häufig angewandt. Während der Umsetzung musste eine Möglichkeit gefunden werden den Schwellwert automatisch zu bestimmen. Häufig wird dazu der mittlere Grauwert eines Bildes verwendet. Die Trennung von Vorder- und Hintergrund ist dabei aber oft nicht optimal. Es wurde deshalb ein statistisches Verfahren implementiert. Das Verfahren von Otsu verwendet zur Schwellwertbildung statistische Hilfsmittel um den Schwellwert besser anzupassen. Zur weiteren Bildverbesserung wurden letztlich noch morphologische Operationen implementiert. Um Bildoperationen auf Gene abzubilden wird diesen ein Index zugewiesen. Das Listing 3.1 zeigt eine Auswahl der wichtigsten Operationen und deren Index.

<i>Sobelfilter</i>	=>	<i>0</i>
<i>Laplace-Filter</i>	=>	<i>1</i>
<i>Mittelwertfilter</i>	=>	<i>2</i>
<i>Gauß-Filter</i>	=>	<i>3</i>
<i>Segmentierung</i>	=>	<i>4</i>

Listing 3.1 Wichtige Operationen mit Index

### 3.2 Erzeugung von Individuen

Die Erzeugung verschiedener Individuen für eine Anfangspopulation ist für die weitere Implementierung sehr wichtig. Die Individuen müssen sehr vielfältig sein. Die Vielfältigkeit betrifft neben der Länge der Chromosomen die verschiedenen Kombinationen aus der Auswahl an Operationen welche zur Verfügung stehen. Ein Gen wird dabei durch den Index einer Bildoperation repräsentiert. Das Vorkommen von gleichen Individuen muss vermieden werden. Als Parameter sind bei der Erzeugung die maximale Anzahl an Genen und die Anzahl der Filter entsprechend zu wählen. Über eine Zufallsvariable werden die Genlängen festgelegt. Bei der Erzeugung des ersten Individuums liegt diese bei 1. Nach jeder Erzeugung wird per Zufall entschieden ob die Genlänge erhöht wird, bis die Maximallänge erreicht ist. Über die Normalverteilung erhält man gut verteilte Populationen. Population und Individuen werden in entsprechenden Klassen implementiert und über Objekte angesprochen.

```
Individuum : 9  
Individuum : 5  
Individuum : 09  
Individuum : 762  
Individuum : 6197  
Individuum : 20834  
Individuum : 84750
```

Listing 3.2 Erzeugung von Individuen mit Kettenlänge 5 und Filteranzahl 10 (0-9)

### 3.3 Fitnessfunktion

Jedem Individuum wird ein Startbild übergeben. Aufgrund seiner Gene erzeugt es ein Ergebnisbild welches dann ein fester Bestandteil des Individuums wird. Die sich daraus ergebene Distanz wird nach dem Verfahren aus Kapitel 2 berechnet. Die Distanz wird dann letztlich als Fitness gewählt.

### 3.4 Genetische Anpassung

Als erster Schritt wird bei der genetischen Anpassung die Rekombination durchgeführt. Über die Turnirselektion werden vorher zwei Individuen ausgewählt die als Eltern deklariert werden. Pro Turnier innerhalb der Population sollte ein Sieger ermittelt werden.

Sind genügend Elternteile vorhanden kann die Rekombination folgen. Die Anzahl der Kindgene wird nach der Vorschrift aus Kapitel 2 berechnet. Die Kreuzung der Elterngene erfolgt dann nach dem *One-Point-Crossover*. Listing 3.3 zeigt die Kreuzung verschiedener Individuen. Jedes neu erzeugte Individuum wird nach der Rekombination in eine neue Population aufgenommen welche die nächste Generation bildet.

<i>Neues Kind : 0150</i>	<i>Vater : 4850</i>	<i>Mutter : 0136</i>
<i>Neues Kind : 1349</i>	<i>Vater : 03849</i>	<i>Mutter : 138</i>
<i>Neues Kind : 550</i>	<i>Vater : 4850</i>	<i>Mutter : 50</i>
<i>Neues Kind : 46</i>	<i>Vater : 746</i>	<i>Mutter : 4</i>

Listing 3.3 Kreuzung verschiedener Individuen

Nach der Erzeugung der nächsten Generation wird auf diese Population die Mutation angewandt. Hierbei werden wie schon beschrieben per Zufall Individuen selektiert und deren Gene an zufällig gewählten Stellen ausgetauscht. Dies erhöht die Vielfältigkeit der Population.

<i>Individuum : 30</i>	<i>Zufallsfilter : 1</i>	<i>mutiertes Individuum : 10</i>
<i>Individuum : 0146</i>	<i>Zufallsfilter : 5</i>	<i>mutiertes Individuum : 0156</i>
<i>Individuum : 4</i>	<i>Zufallsfilter : 2</i>	<i>mutiertes Individuum : 2</i>
<i>Individuum : 46</i>	<i>Zufallsfilter : 0</i>	<i>mutiertes Individuum : 40</i>

Listing 3.4 Mutation verschiedener Individuen

### 3.5 Evolutionärer Algorithmus

Das zentrale Element des Programms bildet der evolutionäre Algorithmus. Hier laufen die bereits umgesetzten Teile der GP sowie der Bildverarbeitung zusammen. Der Algorithmus läuft mit einer äußeren Schleife, welche die Generationen durchläuft. Hierbei wird eine Startpopulation in die festgelegte Anzahl an Folgegenerationen fortgeführt. Eine innere Schleife selektiert und rekombiniert die aktuelle Generation. Letztlich wird noch die Mutation durchgeführt. Damit ist die nächste Generation entstanden, welche im nächsten Schritt als Elternpopulation dient. Neben den beiden Schleifen wird durch Fallunterscheidungen geprüft, ob das Abbruchkriterium erreicht wurde.

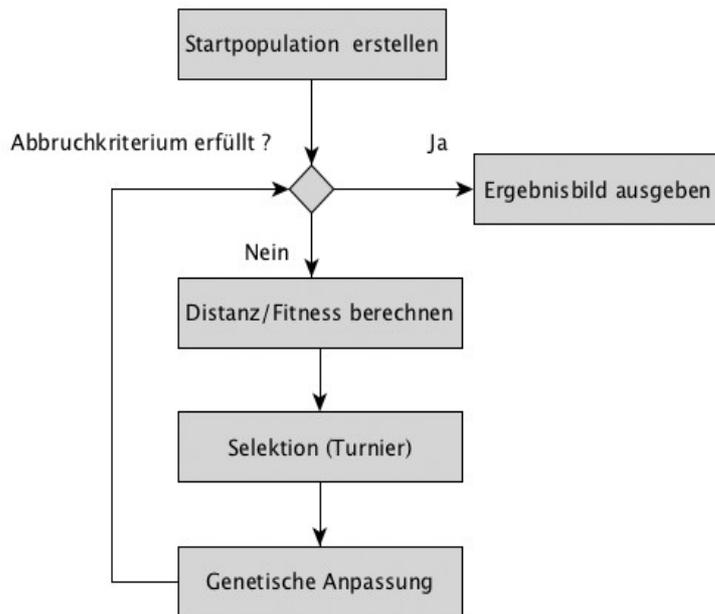


Abb. 3.1 Programmablaufplan „evolutionärer Algorithmus“

## 4 Ergebnisse

In diesem Kapitel sollen zwei Testfälle vorgestellt werden welche im Rahmen der Testphase abgearbeitet wurden. Die beiden Testfälle sind industriellen sowie medizinischen Ursprungs. Sie umfassen die Bereiche Kantendetektion und Segmentierung. Die hier vorgestellten Problemstellungen welche sich aus den Testfällen ergeben sind in industriellen und medizinischen Bereichen häufig anzutreffen.

### 4.1 Testfall Rechtecke

Der Testfall Rechtecke beschäftigt sich mit einer Problemstellung der Kantendetektion. Die Anforderung war zu Anfang so gestellt, dass zur Lösung mindestens zwei Operationen notwendig sind. Das Startbild beinhaltet zwei Rechtecke, welche vor einem grauen Hintergrund abgebildet sind. Die Rechtecke sind räumlich voneinander getrennt und nicht verbunden. Im Endbild sind nur die Kanten dieser Rechtecke sichtbar. Die Problemstellung lautet daher den Weg zur Ermittlung der Kanten nachzuvollziehen.



Abb. 4.1 Startbild Testfall Rechtecke

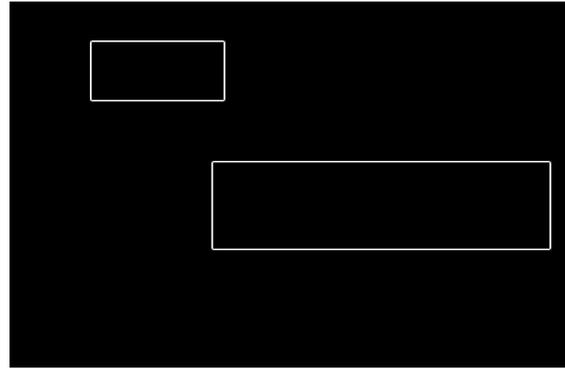


Abb. 4.2 Endbild Testfall Rechtecke

Die Auswahl an Filtern bestand aus Operationen aus den Bereichen Kantendetektion und Rauschunterdrückung wobei manche Filter auch zufällig gewählt wurden. Da Kantendetektoren sehr stark auf Rauschen im Bild reagieren, ist der Einsatz einer Rauschunterdrückung für ein gutes Ergebnis unabdingbar. Bei diesem Testfall sollte zur Kantendetektion vorwiegend der Laplace-Filter verwendet werden. Während des Durchlaufs stellte sich heraus, dass die Kombination aus Laplace-Filter gefolgt vom Gauß-Filter das beste Ergebnis ergab. Da die Rauschanfälligkeit des Laplace-Filters sich als sehr hoch herausstellte, reichte eine einmalige Anwendung des Gauß-Filters nicht aus. Die Anzahl der Durchläufe wurde auf 10 Generationen festgelegt.

Nach drei durchlaufenen Generationen zeigt sich folgendes Bild :

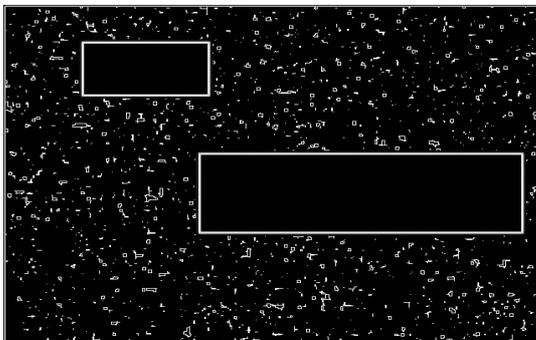


Abb. 4.3 Ergebnisbild nach 3 Generationen

Die Gene des Individuums bestanden dabei aus einer Folge von Laplace-Filter und Gauß-Filter. Deutlich erkennbar bleibt aber noch das Rauschen im Bild. Erst nach der Rekombination in weiteren Generationen konnte die notwendige Kette ermittelt werden. Durch die Kreuzung des Teilergebnisses aus Abb. 4.3 mit Genen, die ebenfalls eine Gauß-Filteroperation enthielten, konnte ein Individuum gefunden werden, welches nur noch eine Distanz von 1,2% aufwies (siehe Abb. 4.4). Der „Lerneffekt“ wird hierbei sehr deutlich. Die Restdistanz von 1,2% ist auf ein Restrauschen im Bild zurückzuführen, das aber so gering ausfällt, dass es nicht sichtbar ist.

Die ermittelte Kette zur Lösung im Durchlauf setzte sich wie folgt zusammen :

*Laplace-Filter => Gauß-Filter => Gauß-Filter*

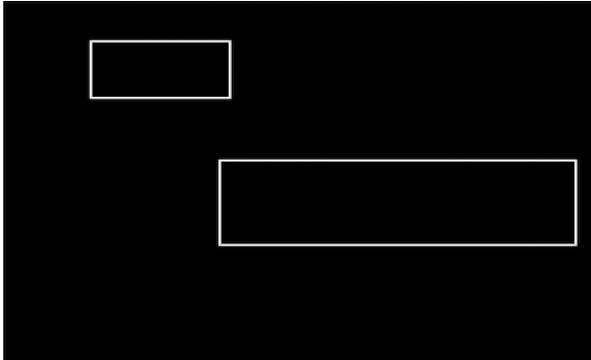


Abb. 4.4 Ergebnisbild nach 10 Generationen

## 4.2 Testfall MRT Bild

Der Testfall stammte aus einem Anwendungsfall aus einer Masterarbeit, die in den Bereich der medizinischen Bildanalyse fiel. Bei den Bildern handelte es sich um medizinische Bilddaten, welche aus einem bildgebenden Verfahren (MRT) stammen. Bei medizinischen Bilddaten müssen oft bestimmte Bereiche gefunden und besonders hervorgehoben werden. Für die Lösung mussten mindestens 3 Filter verwendet werden. Die Filterauswahl war eine Auswahl an Filtern, welche in dem Anwendungsgebiet recht häufig verwendet werden. Dies sind vorwiegend Kantendetektion, Segmentierung und morphologische Operationen. Das Startbild zeigt eine MRT Aufnahme, die den Kopfbereich eines Patienten zeigt. Innerhalb des Schädels ist ein großes Artefakt zu sehen, welchem genauere Betrachtung gewidmet werden soll. Im Ergebnisbild sieht man daher die Reduzierung der Bildinformationen und die gezielte Hervorhebung des Artefaktes. Anwendungen dieser Art werden in der Medizin oft zur Erkennung von Tumoren benötigt. Das Endbild entstand mithilfe eines Snakealgorithmus welcher nicht implementiert wurde. Ziel war es daher sich so gut wie möglich dem Endbild anzunähern.

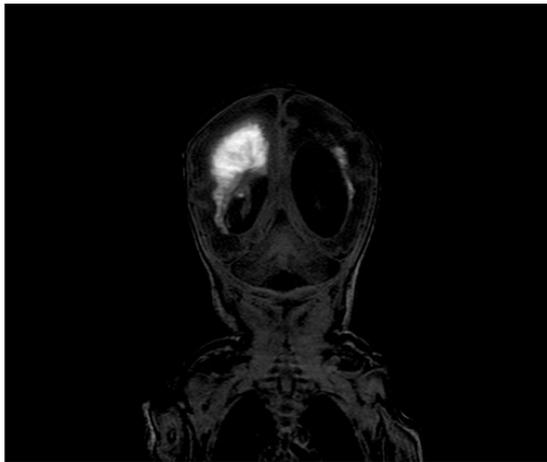


Abb. 4.5 Startbild Testfall MRT-Bild

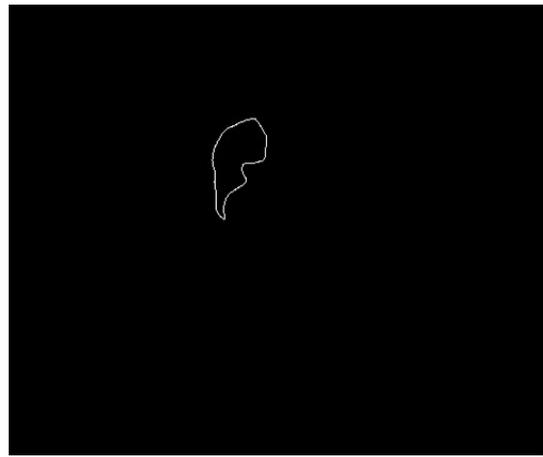


Abb. 4.6 Endbild Testfall MRT-Bild

Die Anzahl der Generationen wurde bei ersten Durchläufen mit 15 festgelegt. Es stellte sich aber heraus, dass 10 Generationen ausreichend sind. Bei diesem Anwendungsfall war vor allem die Schwellwertbildung ein wichtiger Aspekt. Der Bereich deren Umrisse erkannt werden sollen, musste möglichst gut getrennt werden. Hierbei kommt das zuvor vorgestellte Otsu-Verfahren zum Einsatz. Die automatische Schwellwertbildung lässt sich auf den Anwendungsfall sehr gut anwenden (siehe Abb. 4.7). Eine normale (durchschnittliche) Schwellwertbildung würde bei derartigen zusammenhängenden Strukturen keine ausreichende Trennung liefern können. Bei der Kantendetektion erzielte der Sobel-Operator die besseren Ergebnisse (siehe Abb. 4.8). Durch die integrierte Rauschunterdrückung des Sobel-Operators entsteht auch bei komplexen Strukturen ein nahezu rauschfreies Kantenbild.



Abb. 4.7 Anwendung Schwellwert nach Otsu



Abb. 4.8 Anwendung Sobel-Operator

Für ein gutes Ergebnis muss also eine Kombination aus Schwellwertverfahren und Sobel-Operator gebildet werden. Liegt diese Kombination nicht in den Genen der Anfangspopulation, so wird diese über die Rekombination gebildet.

Individuen, welche eine der beiden Operationen in ihren Genen enthalten, setzen sich bei der Selektion im Turnier durch. Allerdings reicht eine Kette bestehend aus Schwellwert und Sobel-Operator alleine nicht aus. Das Ergebnisbild nähert sich zwar dem Endbild an, allerdings ist die gesuchte Struktur noch nicht hervorgehoben (siehe Abb. 4.9). Zudem befinden sich noch kleinere Artefakte im Bild. Diese weisen einen ähnlichen Helligkeitswert wie die gesuchte Struktur auf und können deshalb nicht bei der Segmentierung gefiltert werden.



Abb. 4.9 Sobel-Operator & Segmentierung

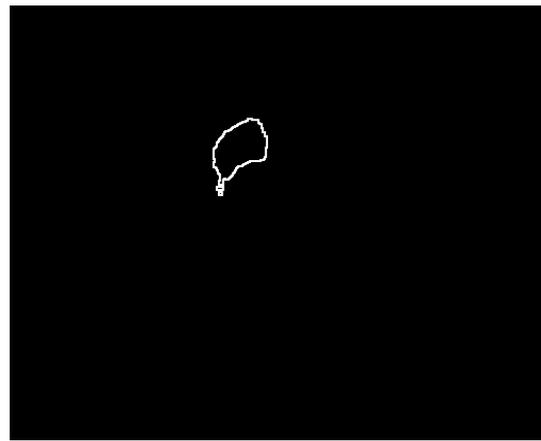


Abb. 4.10 Ergebnisbild - 10 Generationen

Das Ergebnis in Abb. 4.9 liegt im Durchschnitt nach 3-5 Generationen vor. Der Algorithmus findet eine Kombination in welcher durch das Opening kleine Bereiche eliminiert werden. Durch die Anwendung des Opening werden Artefakte herausgefiltert. Die Ergebniskette aus Abb. 4.10 setzt sich wie folgt zusammen :

*Segmentierung => Sobel-Operator => Opening => Segmentierung*

## 5 Fazit und Ausblick

Es konnte gezeigt werden, dass die evolutionäre Bildverarbeitung sich zur Lösung für die im Rahmen des Themas aufkommende Problemstellung sehr gut eignet. Vor allem durch die gezielte Fortführung guter Teilergebnisse, die der evolutionäre Algorithmus durchführt, werden die Vorteile des Verfahrens deutlich. Obwohl die Gebiete der GP und Bildverarbeitung stark voneinander abgegrenzt wirken, lassen sich diese gut miteinander verbinden. Aufgrund des hohen Rechenaufwandes war es früher schwierig die evolutionäre Bildverarbeitung einzusetzen.

Die Laufzeiten des Algorithmus unterschritten bei allen Testfällen im schlechtesten Fall zwei Minuten. Die Datenmenge eines Bildes entsprach dabei der Menge, die auch in der Medizin oder Industrie bei der Aufnahme vieler Bilder vorkommt. Im Rahmen der Arbeit konnten alle Testfälle erfolgreich bearbeitet werden. Vor allem durch den letzten Testfall konnte gezeigt werden, dass das Verfahren sehr effizient reelle Anwendungsfälle aus der Medizin lösen kann.

Für die Zukunft sind Optimierungen und Weiterentwicklungen geplant. Dies betrifft in erster Linie eine Anpassung der Fitnessfunktion bzw. des Distanzmaßes. Die Berechnung der Abweichung zweier Bilder kann mit Hilfe der diskreten Kosinus-Transformation in der Laufzeit verbessert werden. Die im Rahmen der Arbeit verwendeten Distanzberechnungen liefern zwar gute Ergebnisse, sind aber sehr rechenintensiv. Auch im Bereich der Bildoperationen können weitere Laufzeitverbesserungen geschehen. Dies betrifft die Optimierung der Bildverarbeitungsalgorithmen. Ein weiterer Ansatz um das Verfahren erheblich in der Laufzeit zu verbessern ist die Anwendung einer Optimierung mithilfe von Grafikprozessoren. Das Verfahren wurde von Ando und Nagao<sup>1</sup> im Jahr 2009 vorgestellt. Die Anwendung von Grafikprozessoren zur Optimierung des evolutionären Algorithmus könnte die Laufzeit erheblich reduzieren. Vor allem die Anwendung mehrerer Grafikprozessoren und Aufteilung des EA auf mehrere Systeme würde eine deutliche Laufzeitverbesserung bewirken. Auf dieser Basis könnten auch dreidimensionale Bilddaten schnell verarbeitet werden.

## 6 Literatur

- [1] Ebner, M. : „*Evolutionäre Bildverarbeitung*“. Gesellschaft für Informatik e.V. ; 2008
- [2] Ebner, M. : „*Estimating the spectral sensitivity of a digital sensor using calibration*“ ,ACM New York, ISBN: 978-1-59593-697-4; 2007
- [3] Ebner, M. : „*Color Constancy*“, John Wiley & Sons, ISBN: 0470058293; 2007
- [4] Ando J., Nagao T. : „*Fast Evolutionary Image Processing Using Multi-GPUs*“, Yokohama National University, ISBN: 978-953-307-026; 2009

---

<sup>1</sup> Vgl. [4]