

Aktuelles Schlagwort  
**Evolutionäre Bildverarbeitung**

Marc Ebner, Universität Würzburg, Lehrstuhl für Informatik II

Am Hubland, 97074 Würzburg, Germany

ebner@informatik.uni-wuerzburg.de

<http://wwi2.informatik.uni-wuerzburg.de/staff/ebner/welcome.html>

## 1 Motivation

Das visuelle System des Menschen hat eine beachtliche Leistungsfähigkeit erreicht. Im Bereich der maschinellen Bildverarbeitung wurden in den letzten Jahren zwar große Fortschritte erzielt, dennoch sind wir weit davon entfernt, ein künstliches visuelles System aufzubauen, das ähnlich leistungsfähig ist, wie das menschliche System. Da das visuelle System letztlich ein Produkt der natürlichen Evolution ist, ist es sinnvoll, sich einerseits mit dem biologischen Vorbild und andererseits mit dem Prozeß zu beschäftigen, der dieses erzeugt hat. Die Evolution kann auf einem Rechner simuliert werden. So können wir versuchen, ein vergleichbar leistungsfähiges System aufzubauen. Der Ansatz, Probleme aus dem Bereich der maschinellen Bildverarbeitung mit Hilfe der simulierten Evolution zu lösen, wird als evolutionäre Bildverarbeitung bezeichnet (siehe Abb. 1).

## 2 Evolutionäre Algorithmen

Die wesentlichen Merkmale der natürlichen Evolution sind Reproduktion, Variation und Selektion. Aus einer Population von Individuen werden geeignete Individuen selektiert. Hierbei kommt das Darwin'sche Prinzip "Survival of the Fittest" zum Einsatz. Nur den besten Individuen ist es möglich, Nachkommen zu produzieren. In der Natur hängt die Auswahl geeigneter Individuen sowohl von der Zusammensetzung der Population, d.h. von den anderen Individuen ab, als auch von der Umgebung. Die Zahl der Nachkommen eines Individuums während seines Lebens ist die Fitneß des Individuums.

Das Darwin'sche Prinzip können wir nutzen, um technische Optimierungsprobleme zu lösen. Wie in der Natur arbeiten wir mit einer Population von Individuen. Jedes Individuum stellt eine mehr oder weniger gute Lösung unseres Optimierungsproblems dar. Wir dekodieren das genetische Material des Individuums, um eine Lösung, z.B. optimale Parametereinstellungen für ein Optimierungsproblem, zu erhalten. Wie gut ein Individuum das gegebene Problem löst, wird durch eine Fitneßfunktion beschrieben, die durch das Problem definiert ist. Geeignete Individuen werden aus der Population anhand ihrer Fitneß ausgewählt. Dabei wird darauf geachtet, daß überdurchschnittlich gute Individuen stärker ausgewählt werden als Individuen mit einer unterdurchschnittlichen Fitneß.

Auf die ausgewählten Individuen wenden wir in Anlehnung an die Natur eine Reihe von genetischen Operatoren an. Bei Anwendung eines Crossover-Operators wird das genetische Material zweier oder mehrerer Individuen miteinander kombiniert. Bei Anwendung eines Mutations-Operators wird das genetische Material leicht variiert. Die so produzierten Nachkommen werden in eine neue Population integriert. Diese Population wird zur nächsten Generation der Individuen. Wir iterieren diesen Prozeß (Selektion, Reproduktion und Variation) solange, bis ein Abbruchkriterium erfüllt ist, z.B. bis eine maximale Zahl an Generationen erzeugt wurde. Derartige Verfahren werden unter dem Oberbegriff evolutionäre Algorithmen [9] zusammengefaßt. Der Ablauf eines evolutionären Algorithmus ist in Abb. 1 rechts dargestellt.

## 3 Parameter-Optimierung mit evolutionären Algorithmen

Es existieren unterschiedliche Ausprägungen evolutionärer Algorithmen, die sich im Wesentlichen in der Repräsentation der Individuen unterscheiden. Bei den genetischen Algorithmen wird meist mit einer binären Repräsentation der Individuen gearbeitet. Bei den sog. Evolutionsstrategien

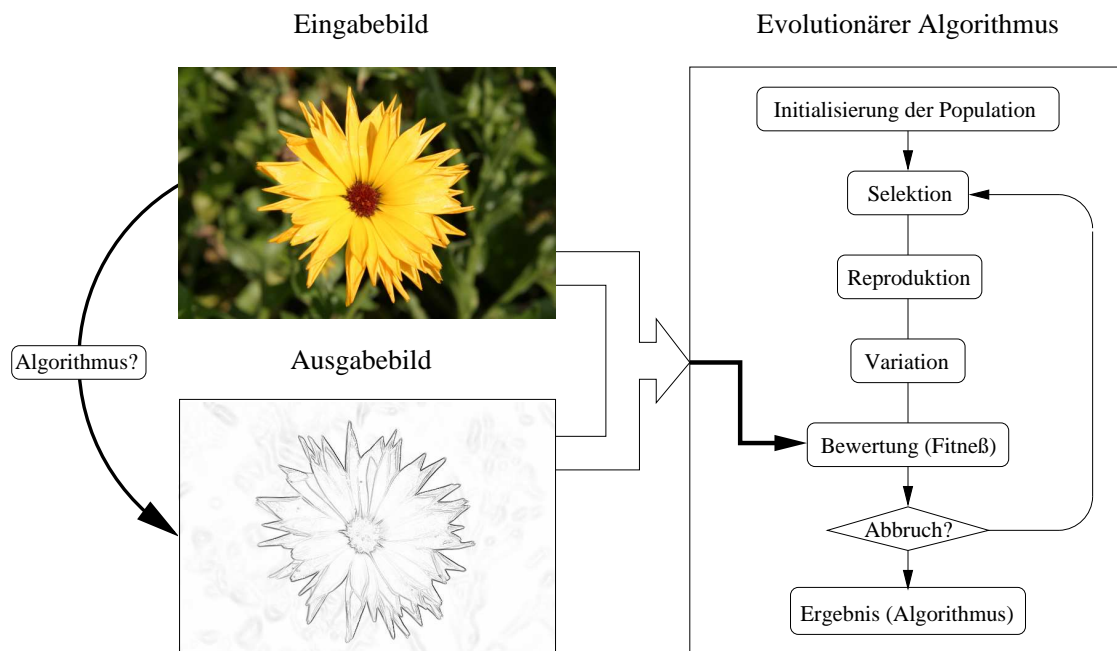


Abbildung 1: Im Bereich der evolutionären Bildverarbeitung wird versucht, ein Bildverarbeitungsalgorithmus mit Hilfe der simulierten Evolution zu erzeugen. Eingabebilder und zugehörige Ausgabebilder bzw. Daten werden zur Bewertung der Individuen herangezogen.

werden Vektoren aus Fließkommazahlen optimiert. Diese Verfahren eignen sich für Probleme der Bildverarbeitung, bei denen lediglich eine Reihe von Parametern, z.B. von Bildverarbeitungsoperatoren, optimiert werden müssen.

Als Beispiel für eine Parameteroptimierung schauen wir uns an, wie wir mit Hilfe einer Evolutionsstrategie die Absorptionscharakteristik eines optischen Systems anhand eines Kalibrierungsbildes bestimmen können [2]. Zunächst nehmen wir mit dem optischen System bestehend aus Optik und digitalem Sensor ein Foto eines Kalibrierungsbildes auf. Das Foto besteht aus mehreren farbigen Flächen, deren Reflektionseigenschaften im Bereich des sichtbaren Spektrums bekannt sind. Jedes Individuum unserer Population beschreibt die Absorptionscharakteristik der drei Sensoren (Rot, Grün und Blau) des digitalen Sensors. Um die Fitneß eines Individuums zu bestimmen, dekodieren wir das genetische Material. Wir erhalten die Absorptionscharakteristik eines durch das Individuum repräsentierten Sensors. Wir simulieren die Absorption des Lichts durch den Sensor, das vom Kalibrierungsbild reflektiert wurde. Je nach der Qualität des Individuums erhalten wir ein Bild, das dem Foto des Kalibrierungsbildes mehr oder weniger ähnlich ist. Je ähnlicher das durch das Individuum erstellte Bild zu dem Foto des Kalibrierungsbildes ist, desto höher die Fitneß des Individuums. Durch die Evolution werden nun solange die Individuen, d.h. die Absorptionscharakteristik des Sensors, variiert, bis ein Individuum gefunden wird, das die Absorptionscharakteristik des optischen Systems beschreibt.

## 4 Evolution von Computerprogrammen für die Bildverarbeitung

Bei vielen Problemen der maschinellen Bildverarbeitung ist nicht bekannt, wie wir welche Operatoren auf ein Bild anwenden müssen, um ein gewünschtes Ergebnis zu erhalten. Der Anwender muß zur Lösung des Problems oft eine Folge von Operatoren zusammenstellen, die das gegebene Problem löst. Er muß also ein Programm schreiben, das aus einer Sequenz von Bildbearbeitungsoperatoren besteht. Zur Lösung derartiger Probleme ist besonders der Bereich der genetischen

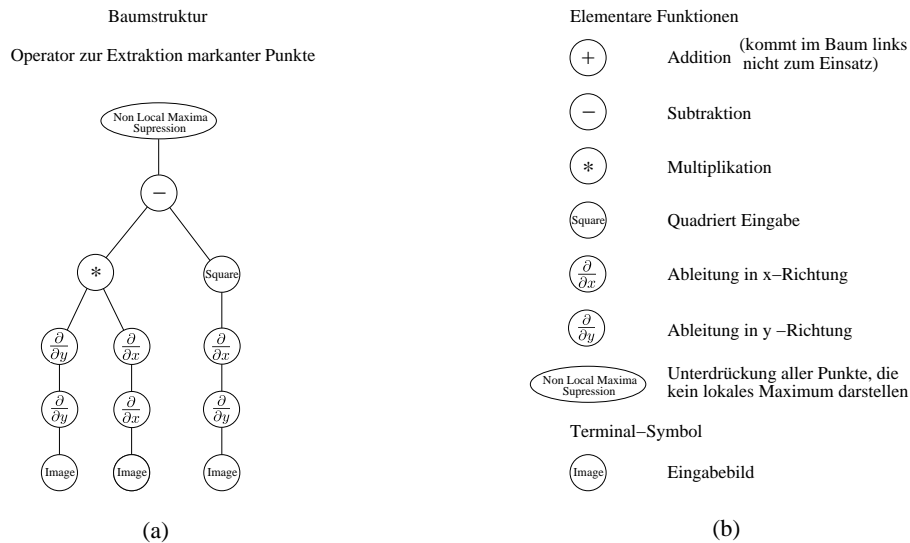


Abbildung 2: In der Bildverarbeitung werden häufig eine Reihe von Operatoren auf ein Bild angewandt, um dieses zu transformieren. (a) Operator zur Extraktion markanter Punkte als Baumstruktur. (b) Elementare Funktionen und Terminal-Symbol, aus denen der Baum aufgebaut ist.

Programmierung relevant [6].

Im Bereich der genetischen Programmierung wird versucht, mit Hilfe der simulierten Evolution automatisch ein Computerprogramm zu finden, das ein gegebenes Problem löst. Bei der baumbasierten genetischen Programmierung werden Baumstrukturen evolviert, die den von Compilern generierten Parse-Bäumen entsprechen. Jedes Individuum besteht aus einem oder auch mehreren Bäumen. Um die genetische Programmierung auf ein Problem der Bildverarbeitung anzuwenden, muß definiert werden, aus welchen inneren und äußeren Knoten ein Baum aufgebaut werden kann. Die inneren Knoten werden als elementare Funktionen, die äußeren Knoten werden als Terminal-Symbole bezeichnet. Als elementare Funktionen können z.B. Operatoren definiert werden, die ein oder mehrere Eingabebilder in ein anderes Bild transformieren, z.B. Addition zweier Bilder, Anwendung eines Kantendetektors, einer morphologischen Operation oder eines Schwellwertoperators auf ein Bild (siehe Abb. 2).

Auch bei der genetischen Programmierung werden die Individuen durch Crossover und Mutation variiert. Bei einer Mutations-Operation wird ein Individuum aus der Population ausgewählt und dann ein Teilbaum dieses Individuums zufällig neu erzeugt. Bei einem Crossover werden zunächst zwei Individuen aus der Population ausgewählt. Aus jedem Individuum wird ein Teilbaum ausgewählt und diese zwischen den Individuen ausgetauscht. Auf diese Weise wandern evtl. relevante Programmstücke von einem Individuum zum anderen bzw. werden dort mit anderen Programmstücken kombiniert. Die elementaren Funktionen sind meist so aufgebaut, daß sie die Ausgabe der anderen elementaren Funktionen als Eingabe verarbeiten können. Auf diese Weise entsteht bei einer Rekombination des genetischen Materials immer ein gültiger Baum als Nachkomme. Es ist jedoch auch möglich, mit elementaren Funktionen zu arbeiten, die nur auf bestimmten Datentypen arbeiten können. In diesem Fall muß bei der Anwendung der genetischen Operatoren darauf geachtet werden, daß nur gültige Individuen erzeugt werden.

## 5 Anwendungen evolutionärer Bildverarbeitung

Eine der ersten Arbeiten im Bereich der evolutionären Bildverarbeitung wurde von Lohmann [5] vorgelegt. Lohmann evolvierte mit einer geschachtelten Evolutionsstrategie Filter, um die in einem Bild vorhandenen Objekte zu zählen. Der evolvierte Filter berechnet die Eulerzahl des Bildes.

Johnson et al. [4] evolvierten mit der genetischen Programmierung komplexe visuelle Routinen, die die Position der linken oder rechten Hand aus einem Bild, das die Silhouette einer Person zeigt, extrahieren.

Treptow und Zell [7] kombinierten einen evolutionären Algorithmus mit dem Adaboost Verfahren, um Bilder, die Gesichter zeigen, von Bildern zu unterscheiden, die keine Gesichter enthalten. Mit Hilfe des evolutionären Algorithmus wurde nach optimalen Filtern gesucht, die an Haar-Wavelets angelehnt sind. Heinemann et al. [3] setzten einen evolutionären Algorithmus im Bereich des RoboCups ein. Beim RoboCup ist das Spielfeld standardisiert und die Roboter müssen innerhalb des Spielfeldes lokalisiert werden. Die von Heinemann et al. verwendeten Roboter sind mit einer omnidirektionalen Kamera ausgestattet. Der evolutionäre Algorithmus optimierte eine Abbildung, die das vom Roboter aufgenommene Bild entzerrt und somit auf Weltkoordinaten transformiert. Auf diese Weise ist der Roboter in der Lage, die Kamera erneut zu kalibrieren, sofern dies nach einer Wartung oder nach dem Transport der Roboter notwendig sein sollte.

Anhand der simulierten Evolution kann auch untersucht werden, wie verschiedene visuelle Fähigkeiten entstanden sein könnten. Ein menschlicher Betrachter ist z.B. in der Lage, die Farbe von Objekten unabhängig von der Farbe der Lichtquelle, die diese beleuchtet, korrekt einzuschätzen. Diese Fähigkeit wird als Farbkonstanz bezeichnet. Mit Hilfe der genetischen Programmierung ist es gelungen, diese Fähigkeit in einem simulierten parallelen visuellen System zu evolvieren [1]. Ferner kann mit der evolutionären Bildverarbeitung untersucht werden, wie sich einfache Operatoren zu komplexen visuellen Operatoren, z.B. zur Merkmalsextraktion zusammensetzen lassen [8].

Die evolutionäre Bildverarbeitung kann in vielen unterschiedlichen Bereichen von der Verarbeitung medizinischer Bilddaten bis hin zu Vision-Systemen für autonome mobile Service-Roboter eingesetzt werden. Betrachten wir ein Beispiel aus dem Bereich Objekterkennung. Wenn wir die evolutionäre Bildverarbeitung verwenden wollen, um einen Algorithmus zu generieren, der ein bestimmtes Objekt extrahiert, so würden wir zunächst Experten in einer Reihe von Bildern die zu extrahierenden Objekte markieren lassen. Dem evolutionären Algorithmus werden dann ein Teil der Bilder zusammen mit Bildern, die das Objekt nicht enthalten, übergeben. Als Fitneßfunktion wird ein Maß definiert, das beschreibt, wie ähnlich die Ausgabe des Individuums, d.h. des evolvierten Algorithmus, zu der Markierung durch die Experten ist. Mit Hilfe der simulierten Evolution wird nun versucht, einen Algorithmus zu finden, der die gleiche Ausgabe, wie die der Experten liefert. Nachdem ein geeigneter Algorithmus gefunden wurde, wird dieser Algorithmus zur Validierung auf dem restlichen Teil der Bilder angewandt und mit den Expertendaten verglichen, um zu überprüfen, ob der Algorithmus allgemeingültig ist und auch unbekannte Daten korrekt verarbeitet. Dieses Prinzip kann auf beliebige zu erkennende Objekte angewandt werden.

Wir können evolutionäre Algorithmen immer dann anwenden, wenn wir in der Lage sind zu sagen, ob ein Algorithmus besser bei der Lösung des gegebenen Problems ist als ein anderer Algorithmus. Die Evolution von Algorithmen für die Bildverarbeitung ist derzeit noch sehr aufwendig, was die erforderliche Rechenzeit anbelangt. Jedes Individuum verarbeitet ein Bild durch Anwendung einer oder mehrerer Bildverarbeitungsoperatoren. Die Rechenzeit, die für die genetischen Operatoren benötigt wird, kann dagegen vernachlässigt werden. Daher ist die erforderliche Rechenzeit grob durch die Zeit gegeben, die wir für die Bewertung eines Individuums benötigen, multipliziert mit der Größe der Population multipliziert mit der Zahl der Generationen, die wir bewerten. Simulationszeiten von mehreren Tagen bis hin zu ein oder zwei Wochen sind dabei keine Seltenheit. In Zukunft könnten Verfahren der evolutionären Bildverarbeitung eingesetzt werden, um adaptive Systeme aufzubauen, die sich an ihre jeweilige Umgebung anpassen und unter schwierigen Lichtverhältnissen bzw. auch mit veränderten Lichtverhältnissen weiterhin funktionieren.

## 6 Aktuelle Ergebnisse der evolutionären Bildverarbeitung

Der erste Workshop zur evolutionären Bild- und Signalverarbeitung fand im Jahr 1999 in Göteborg, Schweden, statt und wird seither jährlich unter der Leitung von Stefano Cagnoni, University of Parma, Italy, organisiert. Im Jahr 2008 wird dieser Workshop bereits zum 11. Mal abgehalten. Er

findet in Neapel, Italien, im Rahmen der EvoStar-Tagung ([www.evostar.org](http://www.evostar.org)) statt, die Konferenzen zur genetischen Programmierung, evolutionäre Algorithmen zur kombinatorischen Optimierung, sowie zur Bioinformatik umfaßt. Im letzten Jahr wurden auf der Tagung in Valencia, Spanien, unter anderem Arbeiten aus den Gebieten Objekterkennung, Bildregistrierung, Gesichtserkennung, Merkmalsextraktion, Bildsegmentierung, Evolution von Filtern zur Erkennung von Defekten bei der Papierproduktion, Reduktion von Rauschen, Bildkompression, Analyse von Ultraschall-Bildern und 3D Modellierung im Bereich der forensischen Identifizierung vorgestellt.

Weitere Arbeiten aus dem Bereich evolutionäre Bildverarbeitung wurden vor kurzem in einer Sonderausgabe der Zeitschrift *Pattern Recognition Letters*, 27(11), Elsevier, 2006, zusammengefaßt. Dort finden sich Artikel zur Evolution eines visuellen Sonars, zur Evolution von Farbkonstanz, zur Bildregistrierung, Bildsegmentierung, Objektklassifizierung bzw. Objekterkennung, zur Erkennung von Gesichtsausdrücken sowie zur Platzierung von Kameras. Eine weitere Sonderausgabe zum Thema evolutionäre Bildverarbeitung wird in der Zeitschrift *Evolutionary Computation*, MIT Press, erscheinen.

## Literatur

- [1] Marc Ebner. Evolving color constancy. *Special Issue on Evolutionary Computer Vision and Image Understanding of Pattern Recognition Letters*, 27(11):1220–1229, 2006.
- [2] Marc Ebner. Estimating the spectral sensitivity of a digital sensor using calibration targets. In *Proceedings of the Genetic and Evolutionary Computation Conference, July 7-11, 2007, London, England*, pages 642–649, New York, 2007. ACM.
- [3] Patrick Heinemann, Felix Streichert, Frank Sehnke, and Andreas Zell. Automatic calibration of camera to world mapping in robocup using evolutionary algorithms. In *Proceedings of the IEEE International Congress on Evolutionary Computation, San Francisco, CA*, pages 1316–1323. IEEE, 2006.
- [4] Michael Patrick Johnson, Pattie Maes, and Trevor Darrell. Evolving visual routines. In Rodney A. Brooks and Pattie Maes, editors, *Artificial Life IV, Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 198–209, Cambridge, Massachusetts, 1994. The MIT Press.
- [5] Reinhard Lohmann. *Bionische Verfahren zur Entwicklung visueller Systeme*. Dissertation, Technische Universität Berlin, Fachbereich 10 Verfahrenstechnik und Energietechnik, 1991.
- [6] Riccardo Poli. Genetic programming for image analysis. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996, Proceedings of the First Annual Conference, July 28-31, 1996, Stanford University*, pages 363–368, Cambridge, Massachusetts, 1996. The MIT Press.
- [7] André Treptow and Andreas Zell. Combining adaboost learning and evolutionary search to select features for real-time object detection. In *Proceedings of the IEEE Congress on Evolutionary Computation, Portland, Oregon*, volume 2, pages 2107–2113. IEEE, 2004.
- [8] Leonardo Trujillo and Gustavo Olague. Synthesis of interest point detectors through genetic programming. In *Proceedings of the Genetic and Evolutionary Computation Conference 2006, Seattle, Washington, July 8-12*, pages 887–894. ACM, 2006.
- [9] Karsten Weicker. *Evolutionäre Algorithmen*. B.G. Teubner Verlag, Wiesbaden, 2007.