Gesichtserkennung mit der Dynamic Link Architektur

Martin Braun
mb13@informatik.uni-ulm.de

Markus Kirschmann
mk10@informatik.uni-ulm.de

1. Einleitung

Bei der Gesichtserkennung aus zweidimensionalen Eingabebildern sind eine Vielzahl von Problemen zu beachten. Meist kann nicht von einer einheitlichen *Position*, *Beleuchtung* und *Skalierung* der Gesichter in den Eingabebildern ausgegangen werden. Zudem resultieren Probleme aus der *natürlichen Beschaffenheit und Form* des Gesichts (Mimik, Gesichtsbehaarung etc.) und aus der *fehlenden Tiefeninformation* in den Bildern (Rotation, Neigung etc.).

Es handelt um ein Klassifikationsproblem, bei dem die Eingabedaten ein und des selben Objektes sehr unterschiedlich sein können. Ein gutes Gesichtserkennungssystem sollte deshalb die differenzierenden Merkmale von unterschiedlichen Gesichtern hervorheben und auf oben genannte Variabilitäten tolerant reagieren. Für Probleme dieser Art haben sich Konzepte Neuronaler Netze als besonders geeignet erwiesen.

2. Die *Dynamic Link Architecture*

Die *Dynamic Link Architecture* (*DLA*)[LVB+93] zeichnet sich gegenüber Neuronalen Netzen, wie z.B. MLP-Netzen[1], durch einige für die Objekterkennung wichtige Eigenschaften aus. Im Gegensatz zu MLP-Netzen durchläuft das Netz in der *DLA* keine gesonderte Trainings- und Klassifikationsphase, sondern es findet *ein* neuronaler Verarbeitungsprozess statt, der sowohl Datenaquisitation, Modifikation der Neuronenverbindungen und die Erkennung der Objekte der Szene ermöglicht.
Die Struktur der *DLA* ermöglicht eine automatische Aufteilung des ganzen Netzes in kleinere funktionale Einheiten, welche für die Erkennung von Objekten und Teilen von Objekten zuständig sind. Dies wird über zeitliche Interaktion der Neuronen ermöglicht, d.h. es existieren Mechanismen welche auf der zeitlichen Korrelation der Ausgabesignale von verbundenen Neuronen beruhen (siehe Abschnitt 2.2).

2.1. Bestandteile der *DLA*

1. **Die Bilddomäne** I **(***image domain***)** Die Bilddomäne (Abb.1) enthält ein zweidimensionales Abbild der Szene mit den zu erkennenden Objekten.
   I besteht aus einem Gitter von Knoten die über den einzelnen Punkten eines rezeptiven Feldes (z.B. eines CCD-Chips) liegen. Jeder Knoten enthält eine Anzahl von Merkmaldetektoren, welche auf spezifische lokale Bildstrukturen reagieren. Knoten sind innerhalb kurzer Distanzen miteinander verbunden. Dieses neuronale Netz kann als Graph mit Knoten-Attributen (Neuronen) und gewichteten Kanten (Synapsen und ihre Verbindungsgewichte) verstanden werden.

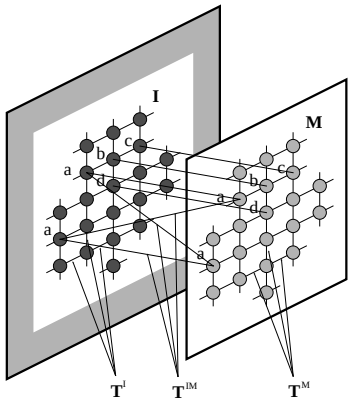

Abbildung 1. Struktur der *DLA*, Bilddomäne I, Modelldomäne M, merkmalerhaltende Verbindungen zwischen Knoten ($a \leftrightarrow a$, $b \leftrightarrow b$, ... aber nicht: $a \leftrightarrow b$, $a \leftrightarrow c$, ...)

2. **Die Modelldomäne** M **(***model domain***)** Die Modelldomäne (Abb.1) enthält eine Sammlung der zu erkennenden, abgespeicherten Vergleichsobjekte. Die Objekte in M sind Kopien von Subgraphen aus I und können aus Eingabebildern extrahiert werden.

3. **Die Verbindungen** T^I**,** T^M**,** T^{IM} Benachbarte Knoten innerhalb einer Domäne sind über kurze Distanzen hinweg verbunden. Neben den Verbindungen T^I zwischen benachbarten Knoten in der Bilddomäne und T^M in der Modelldomäne existieren auch Verbindungen T^{IM} zwischen Knoten der Modelldomäne und Bilddomäne. Verbindungen T^{IM} existieren nur zwischen Merkmaldetektoren des gleichen Typs, nicht jedoch zwischen Merkmalen unterschiedlichen Typs (siehe Abb.1), d.h. die Verbindungen sind merkmalerhaltend.

2.2. Interaktionsmechanismus zwischen Neuronen

Die zeitliche Korrelation zwischen Neuronenausgaben spielt in der *DLA* eine wichtige Rolle. Über die Zeit hin-

[1] MLP = Multi Layer Perzeptron

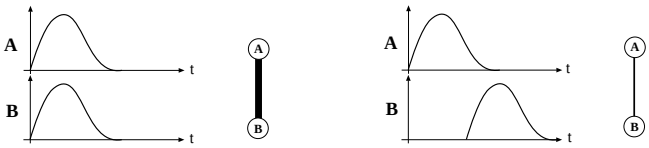

Abbildung 2. Ausgabesignale der Neuronen A und B und Auswirkung auf die Verbindungsstärke zwischen A und B

weg werden die Verbindungen zwischen Neuronen über eine *Rückkopplungsschleife* verstärkt bzw. abgeschwächt:

- Sind die Ausgabesignale zweier Neuronen stark korreliert, führt dies zu einer Stärkung der Verbindung zwischen diesen Neuronen. Dies wiederum führt zur weiteren Erhöhung der Korrelation der Ausgaben dieser beiden Neuronen, d.h. starke Verbindungen verstärken sich noch weiter. (Abb.2, links)
- Analog dazu wird die Verbindung zwischen zwei Neuronen abgeschwächt, wenn die Korrelation zwischen ihren Ausgabesignalen gering ist. Dies führt zur weiteren Verringerung der Korrelation der Ausgaben dieser beiden Neuronen, d.h. schwache Verbindungen schwächen sich weiter ab. (Abb.2, rechts)

2.3. Objekterkennung in der *DLA*

Die Objekterkennung in der *Dynamic Link Architecture* erfolgt in folgenden drei Schritten:

1. **Segmentierung des Bildes I** (Abb.3a) Über den in 2.2 beschriebenen Interaktionsmechanismus werden die Verbindungen T^I zwischen Knoten in I mit ähnlichen Merkmalen verstärkt und gleichzeitig die Verbindungen zwischen Knoten mit stark unähnlichen Merkmalen abgeschwächt. Unter der Annahme, dass einzelne Bildsegmente eine ähnliche Bildstruktur aufweisen, d.h. die extrahierten Merkmalen ähnlich sind, wird so eine erste Aufteilung des Bildes in homogene Regionen erwirkt. Segmentierung ist beispielsweise nützlich, um vorhandene Objekte vom Bildhintergrund zu trennen.
2. **Identifikation und Aktivierung der Knoten in M** (Abb.3b) Um ein Objekt aus M in I zu erkennen, werden die Verbindungen T^{IM} hergestellt, d.h. die Merkmaldetektoren jedes Knotens in M werden mit den Merkmaldetektoren jedes Knotens in I verbunden. Da die Verbindungen zwischen I und M vollständig sind, wird ein ortsinvariantes *Matching* ermöglicht, d.h. das gesuchte Modell kann sich an einer beliebigen Stelle im Bildgraph I befinden, und wird dennoch gefunden.
3. **Reduktion der Verbindungen** (Abb.3c) Zur Lokalisierung des Objektes in I spielen sowohl die Verbindungen zwischen benachbarten Knoten eines Segments bzw. Objektes (T^I, T^M) als auch Verbindungen T^{IM} zwischen I und M eine Rolle. Die Verbindungen T^{IM} werden für Knoten in I und M mit ähnlichen Merkmalen verstärkt und für Knoten mit unähnlichen Merkmalen abgeschwächt. Zugleich wirken Verbindungen zwischen lokal benachbarten Knoten eines Segments in I bzw M um die Topologie eines zusammenhängenden Objektes weitgehend aufrecht zu erhalten.
Auf diese Weise wird ein Orts- und Verzerrungsinvariantes *Matching* vom Objektgraphen im Bild erzielt.

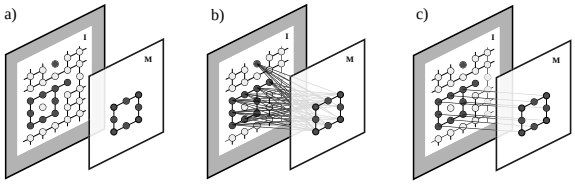

Abbildung 3. Schritte der Objekterkennung in der *DLA*

3. Gesichtserkennung mittels *Elastic Graph Matching*

Da für eine direkte Umsetzung der *DLA* als Neuronales Netz ein sehr hoher und paralleler Rechenaufwand nötig ist, kann sie auf heutigen Digitalrechnern für praktische Zwecke noch nicht effizient eingesetzt werden.
Im folgenden wird deshalb eine aus der *DLA* motivierte Methode zur Objekterkennung (hier speziell zur Gesichtserkennung), vorgestellt, das *Elastic Graph Matching* [WFKvdM97]. Die Graphen der Bild- und Modelldomäne enthalten weniger Knoten als bei der neuronalen *DLA* und enthalten Repräsentationen ausgesuchter Gesichtsmerkmale auf mehreren Auflösungsebenen (Umsetzung mit Gabor-Filtern, siehe Abschnitt 3.1). Der Rechenaufwand reduziert sich dank dieser Vereinfachungen und ermöglicht effiziente Gesichtserkennung auf heutigen Rechnerarchitekturen.

3.1. Gabor Filter als Merkmaldetektoren

Gabor-Filter eignen sich besonders gut, um Informationen über lokale Richtung und Frequenz der Bildstruktur aus natürlichen Bildern zu gewinnen. Sie sind robust gegenüber verschiedenen Beleuchtungssituationen, da nur die Änderung der Helligkeit, nicht jedoch deren Absolutwert, Auswirkungen auf die Antwort des Filters hat.

Ein Gabor-Filter lässt sich über zwei Parameter auf eine bestimmte Vorzugsfrequenz ($f = \frac{2\pi}{\lambda}$) und -richtung (ϕ) justieren. Mit einer Gruppe unterschiedlich parametrisierter Filter lässt das ganze Frequenz- und Richtungsspektrum an im Bild vorkommenden Strukturen abdecken. Auf einer eher konstanten Fläche mit wenig Variation (z.B. Wange) reagieren die hochfrequenten Gabor-Filter weniger stark, als an einer Stelle mit hoher, stark lokalisierter Helligkeitsänderung (z.B. Pupille).

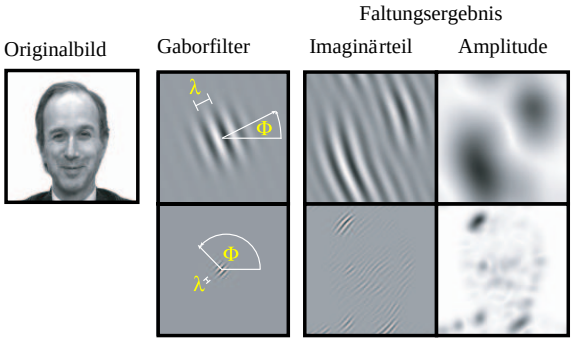

Abbildung 4. Originalbild, Gabor-Filter (sin-Anteil) mit zwei unterschiedlichen Frequenzen und Orientierungen (2.Sp.v.l), Ergebnisse der Faltung des Originalbildes mit dem sin-Filter (2.Sp.v.r) und Amplitude aus den Antworten des sin- und cos-Filters (1.Sp.v.r)

Ein komplexer Gabor-Filter besteht aus einem Realteil (cos-Anteil) welcher auf Bildstrukturen mit gerader Symmetrie reagiert und aus einem Imaginärteil (sin-Anteil) (siehe Abb.4) welcher auf Bildstrukturen ungerader Symmetrie reagiert. Durch das Filtern eines Bildes mit einem Gabor-Filter wird das Bild in eine Darstellung aus komplexen Komponenten überführt, die sich als Punkte in einem Polarkoordinatensystem auffassen lassen. Der Betrag dieser Komponenten ergibt die Amplitudendarstellung eines Bildes (siehe Abb.4, 1.Sp.v.r), der Winkel zwischen Real- und Imaginärteil ergibt die Phasendarstellung. Die Amplitudendarstellung eines Gabor-gefilterten Bildes ist eine glatte Funktion und bietet daher eine gewisse Robustheit gegenüber Translation, Rotation etc., was sich z.B. beim Vergleich zweier leicht gegeneinander verschobener Bilder als vorteilhaft erweist.
Die Phase hingegen schwankt im Gegensatz zur Amplitude schon bei geringen Ortsveränderungen stark. Diese Information kann beim Vergleich zweier gegeneinander verschobener Bilder zur Feinausrichtung verwendet werden.

Die Antworten einer Gruppe Gabor-Filter (5 Orientierungen, 8 Frequenzen, d.h 40 Filter insgesamt) an einem Bildpunkt können als Merkmalsvektor zusammengefasst werden und wird als *Jet* bezeichnet (siehe Abb5).

3.2. Vergleich von *Jets*

Um die Merkmale zweier Gesichter miteinander vergleichen zu können, definieren die Autoren Ähnlichkeitsmaße für *Jets*. Das einfachste Ähnlichkeitsmaß basiert auf der Amplitudeninformation der Filterantworten. Dieser Ansatz resultiert in einer glatten Funktion, so dass ein *Jet* im Eingabebild mit relativ einfachen Suchmethoden (Gradientenabstieg) grob lokalisiert werden kann.
Ein weiteres Ähnlichkeitsmaß, bezieht auch die Phaseninformation der *Jets* mit ein. Dieses wird nach der groben Lokalisierung eines *Jets* im Eingabebild zur Feinausrichtung verwendet, da die Phase im Gegensatz zur Amplitude schon bei geringer örtlicher Abweichung stark variiert.

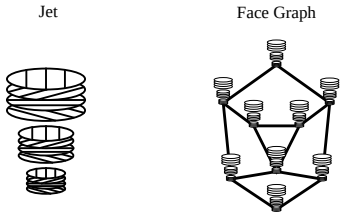

Abbildung 5. *Jets* **und deren Komposition zu einem** *Face Graph*

3.3. *Face Graphs*

Ein Gesicht wird als attributierter Graph repräsentiert und als *Face Graph* bezeichnet. Jeder Knoten in einem *Face Graph* enthält einen *Jet*, der die Struktur eines markanten Merkmals im Gesicht, z.B. Auge, Nase oder Mundwinkel wiederspiegelt (siehe Abb.5).
Die Kanten des Graphen enthalten die Abstände zwischen den jeweiligen Merkmalen.

3.4. *Face Bunch Graphs*

Zur schnellen Gesichtserkennung werden die einzelnen *Face Graphs* zu einem Mengen-Graph, dem *Face Bunch Graph* (*FBG*) zusammengefasst. Der *FBG* enthält die Gesichtsmerkmale (*Jets*) möglichst vieler unterschiedlicher Gesichtstypen (z.B. männlich, weiblich, Augen auf/zu, verschiedene Gesichtsausdrücke, mit/ohne Bart, mit/ohne Brille, leicht gedrehte Ansichten des gleichen Gesichts, etc.). Der *FBG* dient beim *Elastic Graph Matching* (siehe Abschnitt 3.6) zur Lokalisation von Gesichtsmerkmalen im Eingabebild. Dazu wird zu einem Punkt im Bild der am besten passende Jet aus dem FBG gesucht, welches als *local expert* bezeichnet wird (siehe Abschnitt 3.6).

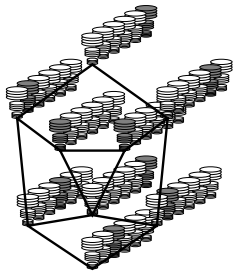

Abbildung 6. Ein *Face Bunch Graph* **(*FBG*), ,,dunkle" Jets stellen die** *local experts* **dar**

3.5. Aufbauen eines *FBG*

Neue *Face Graphs*, die sukzessive zum *FBG* hinzugefügt werden, werden in einem halbautomatischen Prozess erstellt. Der Benutzer muss anfangs die markanten Gesichtspunkte für das System im Bild markieren. Später, wenn der *FBG* genügend viele unterschiedliche Gesichtsmerkmale gespeichert hat, können neue *Face Graphs* automatisch mittels *Elastic Graph Matching* aus den Eingabebildern extrahiert werden. Auf diese Weise lässt sich dann automatisch eine Gesichter-Datenbank erzeugen.

3.6. Gesichtsidentifikation mittels *Elastic Graph Matching*

Um ein bestimmtes Gesicht in I zu erkennen, wird aus dem *FBG* ein ,,Durchschnittsgraph". errechnet, d.h. die Knoten enthalten Durchschnittswerte der *Jets* der Knoten des *FBGs*. Die Kanten werden mit Durchschnittsabständen markiert. Mit dem ,,Durchschnittsgraph" wird also ein mittleres Gesicht über alle *Face Graphs* des *FBG* berechnet.

Der eigentliche Extraktionsprozess eines *Face Graphs* aus dem Eingabebild läuft dann in folgenden Schritten ab:

1. **Grobpositionierung des Durchschnittsgraphen:** Der Graph wird in 4 Pixel Schritten über das Eingabe-Bild I (128x128 Pixel) geschoben. An jeder Position werden alle *Jets* des Graphen mit den *Jets* des Bildes an den entsprechenden Positionen verglichen. Über ein auf der Amplitude der Jets beruhendes Ähnlichkeitsmaß, wird die Position maximaler Übereinstimmung bestimmt. Anschliessend wird der Graph an der so gefundenen Position nochmals mit dem gleichen Verfahren und 1 Pixel Schrittweite genauer positioniert.

2. **Genauere Positionierung und proportionale Grössenanpassung:** Im Folgenden wird anstatt des Durchschnittsgraphen, der eigentliche *FBG* verwendet. Dieser wird jeweils in einer, unter Beibehaltung der Seitenverhältnisse, um 18% vergrösserten und verkleinerten Version, an 4 Positionen um die in Schritt 1 bestimmte Position, mit dem Bild verglichen - es werden also 8 Graphen-Vergleiche durchgeführt. Hierfür werden jeweils die unterschiedlichen *Jets* eines Knotens des *FBG* mit dem *Jet* an der entsprechenden Position des Bildes verglichen. Für jeden Knoten wird der *Jet* des *FBG* ausgewählt, der dem Bildinhalt am ähnlichsten ist, d.h der *local expert* bestimmt. Nun wird die Verschiebung der Knoten dieser 8 Graphen mithilfe der Phaseninformation der *Jets* geschätzt und die Größe und Position der Graphen angepasst, sodass jeder der 8 Graphen möglichst gut mit dem Bild übereinstimmt. Der passendste Graph dient als Ausgangsgraph für Schritt 3.

3. **Genauere Positionierung und freie Grössenanpassung:** Der in Schritt 2 gefundene Graph wird nun erneut skaliert und positioniert, jedoch ohne Erhalten der Seitenverhältnisse.

4. **Lokale Positionierung der einzelnen Knoten:** Die einzelnen Knoten des Graphen werden nun zufällig in einem kleinen Ausschnitt um die in Schritt 3 bestimmte Position gesetzt und mit dem Bildinhalt verglichen um lokale Verzerrungen auszugleichen. Bei diesem Schritt werden zum ersten mal auch die Kantengewichte in die Ähnlichkeitsberechnung miteinbezogen, um eine zu starke Verzerrung des Graphen zu vermeiden.

Der so aus einem Probe-Bild extrahierte *Face Graph* kann nun mit den Modellen der Gesichts-Datenbank verglichen werden, und das ähnlichste Gesicht kann bestimmt werden. Überschreitet die Ähnlichkeit einen bestimmten Schwellwert kann somit ein Gesicht eindeutig identifiziert werden.

3.7. Experimente auf den Datenbanken

Um die Leistungsfähigkeit des vorgestellten Gesichtserkennungssystems zu ermitteln und das System mit anderen Systemen vergleichen zu können, wurden Experimente auf Referenzdatenbanken durchgeführt. Eine dieser Datenbanken ist die FERET Gesichter Datenbank vom US Army Research Labratory. Sie besteht aus Bildern (256x384 Pixel) verschiedener Ansichten von Gesichtern (Frontalansicht, Halb- und Vollprofil) und Varianten des gleichen Gesichts mit unterschiedlichem Gesichtsausdruck. Für die Tests wurden Modell- und Probedatensätze mit jeweils 250 Bildern gewählt.

Die Tests bestätigen eine gute Leistungsfähigkeit des Systems bei der Erkennung von Gesichtern der gleichen Ansicht. Variationen im Gesichtsausdruck wirken sich kaum negativ auf die Erkennungsraten aus. Auch unterschiedliche Skalierung und Position der Gesichter in den Eingabebildern werden vom System gut toleriert. Zu starke Rotation der Gesichter (bei Kopfdrehung) führen jedoch zu einer starken Abnahme der Erkennungsleistung. Werden z.B. Halbprofile (ca 45°) mit Frontalansichten (0°) verglichen, sind die Erkennungsraten gering. Es wurde gezeigt, dass die Performanz des Systems bei Kopfrotationen bis zu 22° noch sehr gut ist.

Literatur

[LVB+93] M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Würtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42:300–311, 1993.

[WFKvdM97] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. In G. Sommer, K. Daniilidis, and J. Pauli, editors, *Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP'97, Kiel*, number 1296, pages 456–463, Heidelberg, 1997. Springer-Verlag.