

Implementatie van een Dynamic Time Warping algoritme

Citation for published version (APA):

Borger, M. W. M. (1990). *Implementatie van een Dynamic Time Warping algoritme*. (IPO rapport; Vol. 743). Instituut voor Perceptie Onderzoek (IPO).

Document status and date:

Gepubliceerd: 10/05/1990

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Instituut voor Perceptie Onderzoek
Postbus 513, 5600 MB EINDHOVEN

Rapport no. 743

10 V 1990

Implementatie van een Dynamic
Time Warping algoritme

M.W.M. Borger

Faculteit der Technische Natuurkunde
Technische Universiteit
Eindhoven

Implementatie van een Dynamic Time Warping algoritme

M.W.M. BORGER

13 januari 1990

Verslag van het stagewerk
uitgevoerd van december 1989 tot januari 1990
op het Instituut voor Perceptie Onderzoek
Begeleiders: Berry Eggen en Werner Verhelst

Abstract

Based upon the many experiences in the field of speech recognition a dynamic time warping algorithm has been implemented. Its purpose is to align subsequent frames of several utterances of the same sentence by the same speaker as good as possible. Though DTW-based recognizers were designed to match short utterances only, we have applied the DTW principle to solve our problem of aligning longer utterances. The warping capability of the resultant program is not limited to the conditions stated above (one speaker, one sentence). Any two utterances can be warped. The performances have been tested using utterances of the same sentence by different speakers as well as artificial time-axis manipulated versions of a single utterance. The performances of the algorithm were found to be satisfactory.

Samenvatting

In dit rapport wordt verslag gedaan van een stage die verricht is bij de Akofon-groep van het Instituut voor Perceptie Onderzoek (IPO). Doel was een Dynamic Time Warping (DTW) algoritme te implementeren, waarmee verschillende uitingen van dezelfde zin door dezelfde spreker qua tijdsverloop op elkaar kunnen worden afgebeeld. Hierbij werd uitgegaan van de vele publikaties van onderzoeken op het gebied van DTW-gebaseerde woordherkenning. Hoewel de op DTW gebaseerde woordherkenners zich beperken tot het warpen van korte uitingen (woorden) zijn de DTW principes hier toegepast om langere uitingen (zinnen) op elkaar af te beelden. Het resulterende programma is niet beperkt tot het warpen van verschillende uitingen van dezelfde zin door dezelfde spreker. Ook uitingen van verschillende sprekers kunnen worden gewarped. Het geïmplementeerde programma is getest met behulp van kunstmatig tijdsgemanipuleerde versies van een korte uiting en uitingen van zinnen door verschillende sprekers. Hoewel handmatige controle van de testresultaten binnen het gegeven tijdsbestek niet mogelijk was, bleken ze globaal goed aan de verwachtingen te voldoen.

Inhoud

1	Inleiding	1
2	Theorie	2
2.1	De warpingfunctie	2
2.2	Beperkingen t.a.v. de warpingfunctie	4
2.3	Bepaling van de warpingfunctie	5
2.4	Toepassing van extra restricties	9
2.4.1	Slope constraints	9
2.4.2	Window	10
2.4.3	Penalties	13
2.5	Keuze van de norm	15
2.5.1	Algemeen	15
2.5.2	De cepstrale norm	16
2.5.3	Bandpass liftered cepstrale afstand	17
3	Implementatie	20
3.1	Korte beschrijving	20
3.2	De interactie met de gebruiker	20
3.3	Uitvoer	23
4	Testen van het programma	25
4.1	Kunstmatige tijdas manipulatie	27
4.2	Realisaties van eenzelfde zin door verschillende sprekers . .	31
4.3	Conclusies	34
	Literatuur	35
A	Machine-uitvoer van de programmatests	36

Hoofdstuk 1

Inleiding

Dat verschillende uitingen van hetzelfde woord een verschillend tijdsverloop hebben is een bekend probleem op het gebied van woordherkenning. Het kan worden opgelost met behulp van de in dit onderzoeksgebied ontwikkelde techniek Dynamic Time Warping (DTW), waarmee spraakuitingen qua tijdsverloop op elkaar kunnen worden afgebeeld. Ook in breder verband, los van een herkenningstaak, kan DTW een nuttig instrument ten behoeve van het spraakonderzoek vormen.

In deze stage is een DTW algoritme geïmplementeerd, dat in staat is lange spraakuitingen qua tijdsverloop op elkaar af te beelden. Het gaat hierbij in eerste instantie om verschillende realisaties van dezelfde zin door dezelfde spreker. Om tot een goede keuze van parameterwaarden te komen, en om een goed beeld te vormen van de consequenties van bepaalde keuzes, is gebruik gemaakt van de vele publicaties van onderzoeken die direct of indirect met DTW te maken hebben.

In hoofdstuk 2 is uiteengezet hoe verschillen in tijdsverloop kunnen worden weergegeven door een warpingfunctie en hoe deze functie kan worden bepaald. Er wordt een basisalgoritme opgesteld, gevolgd door een bespreking van mogelijke toevoegingen. Vervolgens wordt dieper ingegaan op de keuze van de gehanteerde afstandsmaat. In hoofdstuk 3 is beschreven hoe het algoritme is geïmplementeerd. Met name de interactie met de gebruiker en de uitvoer van het programma krijgen hier de aandacht. Tenslotte worden in hoofdstuk 4 de prestaties van het programma aan de hand van enkele tests besproken.

Hoofdstuk 2

Theorie

2.1 De warpingfunctie

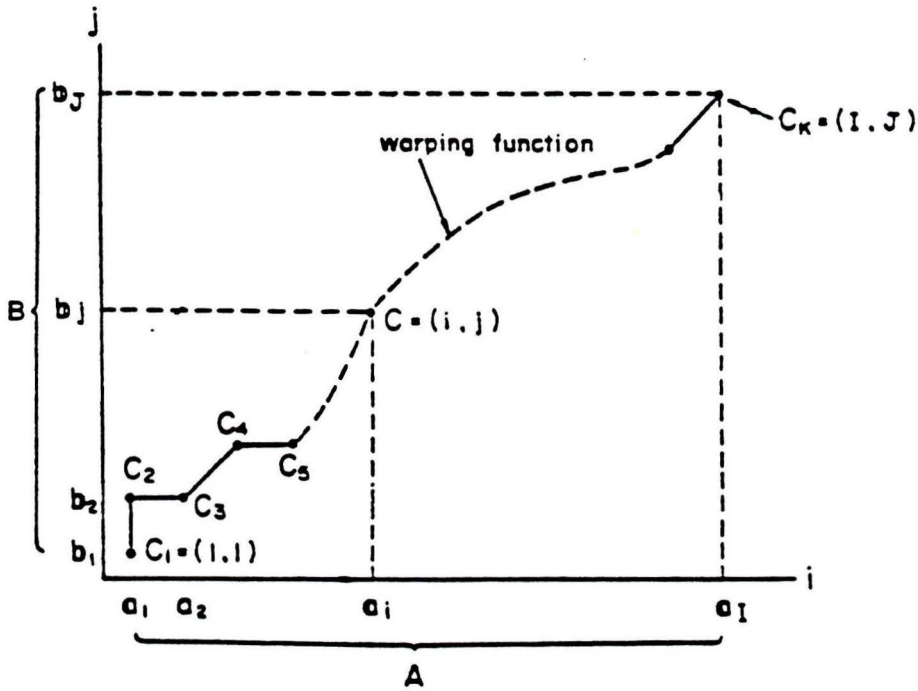
Uitgaande van analyse en zuinige codering op basis van het zogenaamde bron-filter model kan een stukje spraak worden gerepresenteerd door een reeks vectoren

$$A = \vec{a}_1, \vec{a}_2, \dots, \vec{a}_i, \dots, \vec{a}_I$$

waarin de index de tijd representeert. Op dezelfde wijze leidt een tweede realisatie van dezelfde zin tot een reeks vectoren

$$B = \vec{b}_1, \vec{b}_2, \dots, \vec{b}_j, \dots, \vec{b}_J$$

Ten gevolge van fluctuaties in de spreeknelheid ontstaan niet-lineaire verschillen in het tijdsverloop van beide uitingen, zodat een direct vergelijk tussen beide reeksen vectoren niet mogelijk is. De verschillen in tijdsverloop kunnen worden weergegeven door een reeks punten $\mathbf{c}(k) = (i(k), j(k))$ vast te stellen, zodanig dat $\vec{a}_{i(k)}$ zo goed mogelijk overeen komen met $\vec{b}_{j(k)}$. De reeks $\{\mathbf{c}(k)\}$ representeert dan een functie die bij benadering de tijdassen van beide spraakuitingen op elkaar afbeeldt, en wordt daarom de warpingfunctie genoemd. Grafisch kan de warpingfunctie worden weergegeven als in figuur 2.1.



Figuur 2.1: Grafische representatie van de warpingfunctie. Langs de horizontale as zijn de tijdstippen i uitgezet die overeen komen met de vectoren van de eerste spraakuiting. Langs de verticale as hetzelfde voor de tweede uiting. Via de warpingfunctie kan nagegaan worden hoe de tijdstippen op elkaar afgebeeld worden.

Als maat voor het verschil tussen twee vectoren \vec{a}_i en \vec{b}_j wordt een afstandsmaat

$$d(c(k)) = d(\vec{a}_{i(k)}, \vec{b}_{j(k)}) = \|\vec{a}_{i(k)} - \vec{b}_{j(k)}\|$$

gehanteerd, waarvan de exacte vorm nog vrij te kiezen is. De sommatie

$$G(A, B) = \sum_{k=1}^K d(c(k)) \quad (2.1)$$

van alle afstanden langs de warpingfunctie wordt de globale afstand genoemd. Een minimale waarde wordt bereikt als de warpingfunctie $F = \{c(k)\}$ optimaal bepaald is. De minimale globale afstand

$$D(A, B) = \min_F \sum_{k=1}^K d(c(k)) \quad (2.2)$$

wordt ook wel tijdgenormaliseerde afstand genoemd, aangezien zij minimaal is ten opzichte van variaties in het verschil in tijdsverloop tussen beide spraakuitingen. Haar van nul verschillende waarde is het gevolg van resterende verschillen na eliminatie van tijdasdeformaties. We zullen in paragraaf 2.3 de warpingfunctie F trachten te vinden uit het minimaliseren van de globale afstand.

2.2 Beperkingen t.a.v. de warpingfunctie

De warpingfunctie is een model voor tijdasfluctuaties in natuurlijke spraak, en moet derhalve de eigenschappen van werkelijk optredende fluctuaties goed beschrijven. Essentiële eigenschappen van spraak zijn o.a. monotonie en continuïteit. Deze kunnen in de vorm van transitieregels opgelegd worden aan de warpingfunctie. Deze regels luiden:

$$i(k-1) \leq i(k) \text{ en } j(k-1) \leq j(k) \quad (\text{monotonie}) \quad (2.3)$$

$$i(k) - i(k-1) \leq 1 \text{ en } j(k) - j(k-1) \leq 1 \quad (\text{continuïteit}) \quad (2.4)$$

Combinatie van beide restricties geeft de volgende mogelijke relaties tussen twee opeenvolgende punten:

$$c(k-1) = \begin{cases} (i(k), j(k) - 1) \\ (i(k) - 1, j(k) - 1) \\ (i(k) - 1, j(k)) \end{cases} \quad (2.5)$$

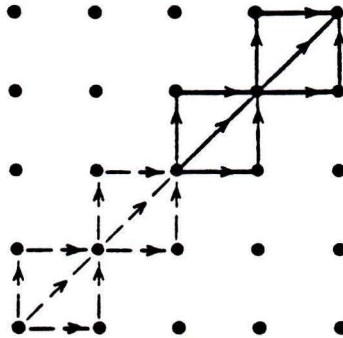
Naast deze transitieregels gelden nog randvoorwaarden die tot uitdrukking brengen dat de warpingfunctie begint in het punt $(1,1)$ en na K stappen eindigt in het punt (I, J) :

$$i(1) = 1, j(1) = 1 \text{ en } i(K) = I, j(K) = J \quad (2.6)$$

Bovendien kunnen nog beperkingen opgelegd worden aan de maximale afwijking van de rechte die begin en eindpunt met elkaar verbindt, alsmede de maximaal toegestane hellingen van de warpingfunctie. Deze beperkingen zijn echter niet essentiële. Ze dienen slechts om rekentijd te sparen en al te onwaarschijnlijke vormen van de warpingfunctie van te voren uit te sluiten. In paragraaf 2.4 zullen deze mogelijkheden nader worden beschouwd.

2.3 Bepaling van de warpingfunctie

We hebben het probleem van het vinden van de warpingfunctie herleidt tot het minimaliseringsprobleem (2.2). Een rechttoe-rechtaan methode om de minimale globale afstand te vinden zou er uit bestaan de globale afstanden langs alle mogelijke paden die begin en eindpunt met elkaar verbinden te berekenen. Dit zou tot een zeer groot aantal berekeningen leiden. Uit (2.5) blijkt dat ieder punt, met uitzondering van de randpunten $(1, j)$ en $(i, 1)$, steeds drie mogelijke buurpunten heeft. Beschouw in het bijzonder het eindpunt (I, J) . Tezamen met zijn buurpunten vormt deze een 2×2 matrix. Er zijn drie mogelijke paden die de punten $(I - 1, J - 1)$ en (I, J) met elkaar verbinden. Evenzo zijn er drie mogelijke paden tussen de punten $(I - 2, J - 2)$ en $(I - 1, J - 1)$. Zodoende zijn er in een 3×3 matrix meer dan $3 \cdot 3$ mogelijke paden van linksonder naar rechtsboven (zie figuur 2.2). Iedere uitbreiding van de matrix met 1 orde levert zo tenminste een factor 3 meer paden op. Veronderstellen we $I < J$, dan is het totaal aantal mogelijke paden groter dan 3^{I-1} , d.w.z. het aantal benodigde berekeningen verloopt exponentieel met de lengte van de spraakuiting.

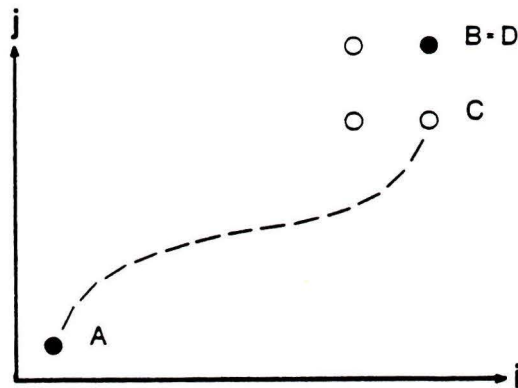


Figuur 2.2: *Het aantal mogelijke paden in een matrix indien we ons beperken tot de diagonaal met zijn directe buurpunten. Dit geeft een exponentieel verloop met de matrixgrootte te zien.*

We kunnen deze stelling als volgt toepassen op ons minimaliseringsprobleem: We identificeren A met het beginpunt $(1, 1)$ en B met het eindpunt (I, J) (zie figuur 2.4). Voorts nemen we voor D het eindpunt en trachten we punt C te bepalen uit de drie mogelijke buurpunten van D , waarbij we formeel veronderstellen dat de minimale globale afstanden tussen A en de drie buurpunten van D bekend zijn. De stelling zegt ons dat C overeen moet komen met die buur, waarvoor geldt dat de lokale afstand tussen de spraakvectoren \vec{a}_I en \vec{b}_J , $d(D)$, vermeerderd met de globale afstand tussen A en C , $G(C)$, minimaal is met betrekking tot C . Punt C volgt dus uit minimalisering van

$$G(D) = d(D) + G(C) \quad (2.7)$$

met betrekking tot C .



Figuur 2.4: *Bepaling van de warping-functie. Indien CD deel uitmaakt van het globaal minimale pad tussen A en B , dan moet de globale afstand tussen A en C minimaal zijn. Punt C komt dus overeen met die buur van D waarvoor de globale afstand tot A minimaal is.*

Na punt C te hebben bepaald hernoemen we $C \rightarrow D$ en passen boven genoemde procedure opnieuw toe om het volgende punt C te bepalen. Uiteraard kennen we de globale afstanden $G(C)$ nog niet en lijkt het probleem

in zichzelf gevangen. Dit lossen we eenvoudig op door niet bij het eindpunt (I, J) te beginnen maar bij $(1, 1)$, en alle matrixpunten als mogelijk punt op het te vinden pad te veronderstellen. Lopen we de gehele matrix af van links naar rechts en per kolom van beneden naar boven, dan speelt elk punt eenmaal de rol van punt D en wordt het globaal minimale pad tot aan ieder punt uit de voorgaande kennis bepaald. Uiteindelijk zal het eindpunt (I, J) aan de beurt zijn en is daarmee het globaal minimale pad tussen begin en eindpunt bekend. Immers we weten dat het eindpunt op het pad ligt en de globale afstanden tot de burens zijn nu bekend.

Bij de toepassing van deze methode ten behoeve van woordherkenning wordt niet het pad zelf, maar alleen de globale afstand langs het pad bepaald. Om achteraf het pad (de warpingfunctie) terug te vinden onthouden we bij ieder matrixpunt welk van de buurpunten (2.7) minimaliseert.

We kunnen de hierboven beschreven methode samenvatten in onderstaand algoritme:

Initialisatie : $G(1,1) = d(1,1)$

$G(i,0) = \infty$

$G(0,j) = \infty$

$G(0,0) = 0$

loop $i = 1$ to I

loop $j = 1$ to J

Dynamic Programming Vergelijking :

$$G(i,j) = \min \left\{ \begin{array}{l} d(i,j) + G(i,j-1) \\ d(i,j) + G(i-1,j-1) \\ d(i,j) + G(i-1,j) \end{array} \right\}$$

end loop j

end loop i

2.4 Toepassing van extra restricties

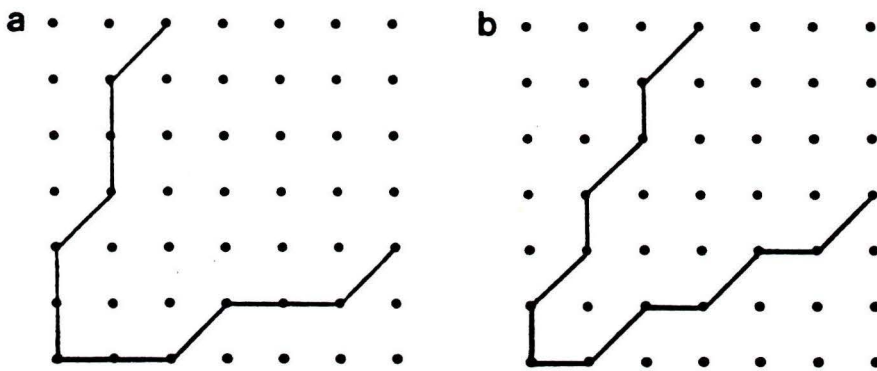
Naast de tot nu toe gebruikte restricties en randvoorwaarden kan met behulp van extra beperkingen de hoeveelheid rekenwerk enigszins teruggebracht worden.

2.4.1 Slope constraints

In woordherkenning is een pad dat vele stappen in horizontale of verticale richting maakt onwaarschijnlijk. De deformatie van de tijdassen is dan immers zeer groot. Om dergelijke paden van te voren uit te sluiten kan een begrenzing aan de lokale helling van het pad opgelegd worden (slope constraint). In navolging van Sakoe en Chiba [1] definiëren we een parameter p die dit weergeeft als volgt:

$1/p =$ het aantal geoorloofde stappen in horizontale danwel verticale richting na een stap in diagonale richting.

$p = 0$ betekent dus geen restrictie. $p = 1$ houdt in dat na iedere horizontale of verticale stap een diagonale moet volgen, d.w.z. de tangens van de helling is begrensd tussen $\frac{1}{2}$ en 2 (zie figuur 2.5).

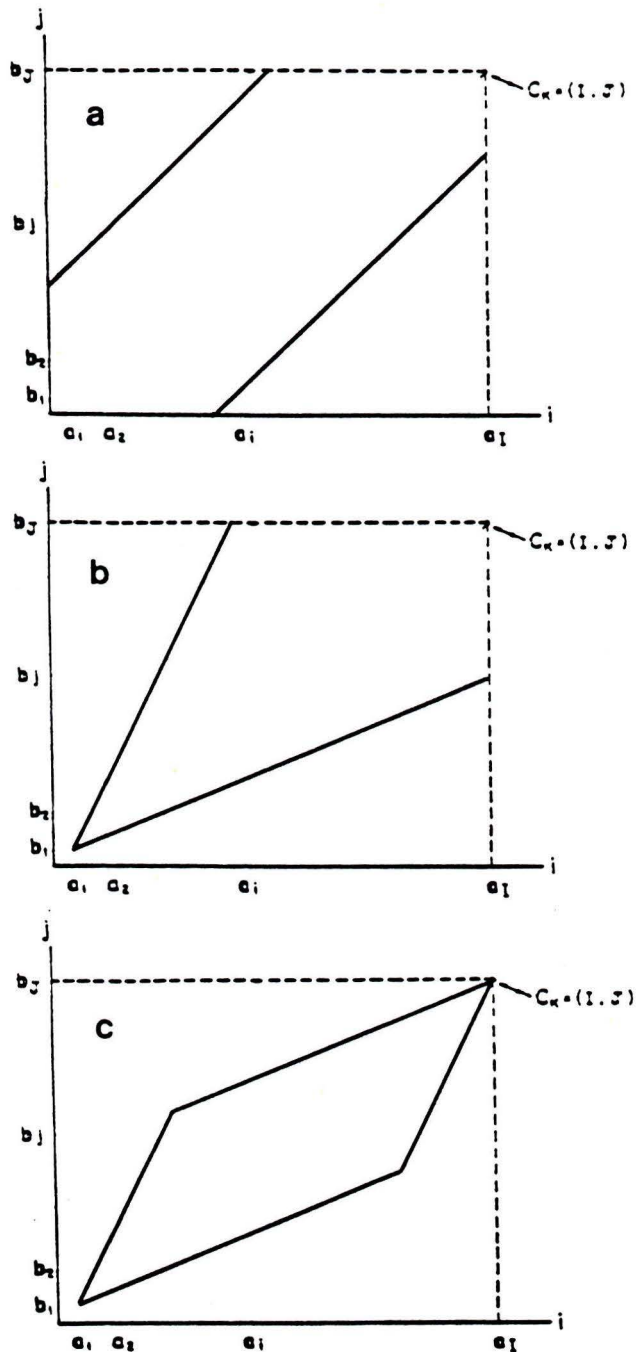


Figuur 2.5: *Illustratie van de betekenis van de parameter p . (a) $p=0.5$ beperkt de richtingscoëfficiënt tussen de waarden $1/3$ en 3 . (b) $p=1$ beperkt de richtingscoëfficiënt tussen de waarden $1/2$ en 2 .*

In het geval $p \neq 0$ kan bovendien geëist worden dat een horizontale en een verticale stap elkaar niet mogen opvolgen. Een dergelijke opeenvolging zou immers betekenen dat de tijdassen direct achter elkaar in twee tegengestelde richtingen sterk gedeformeerd moeten worden, hetgeen onwaarschijnlijk is. Het toepassen van slope constraints heeft in woordherkenning tot een verbetering van de herkenningsscore geleid, d.w.z. het percentage ten onrechte herkende of ten onrechte afgewezen woorden neemt af onder toepassing van slope constraints. Sakoe en Chiba [1] vonden gelet op herkenningsscore voor p een optimale waarde 1. In onze toepassing op zinnen die opzettelijk met verschillende intonatie zijn ingesproken is voorzichtigheid met betrekking tot de toepassing van slope constraints geboden. Verschillen in intonatie leiden onvermijdelijk tot verschillen in bijvoorbeeld klemtoon en rustpauzes, die aanleiding kunnen geven tot grote tijdasdeformaties.

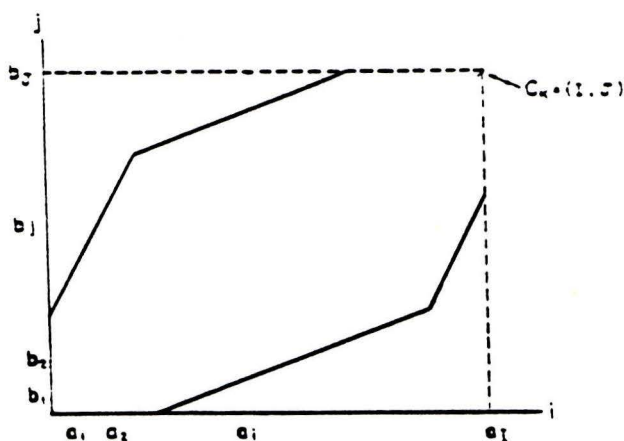
2.4.2 Window

Een voor de hand liggende restrictie is de toepassing van een window waarbinnen het pad zich dient te bevinden. Immers een al te grote afwijking van de rechte die begin en eindpunt met elkaar verbindt is onwaarschijnlijk. Voor de vorm van het window zijn in de literatuur diverse voorstellen gedaan. De meest eenvoudige bestaat uit twee rechten parallel aan de diagonaal [1] (zie figuur 2.6(a)). Een andere mogelijkheid is een begrenzing met de uiterste rechten die volgen uit de toepassing van slope constraints (zie figuur 2.6(b)). Hoewel dit geen expliciete toepassing van een window inhoudt, wordt op deze wijze bij het doorlopen van de matrix toch rekening gehouden met het feit dat een gedeelte daarvan buiten het gebied van de mogelijke paden ligt. Indien de uiterste rechten, zoals die door de slope constraints worden vastgelegd, symmetrisch worden toegepast, zoals door Itakura [2], ontstaat wel weer een expliciet opgelegd window (zie figuur 2.6(c)).



Figuur 2.6: Voorbeelden van windows: (a) Begrenzing door rechten evenwijdig aan de diagonaal. (b) Begrenzing door slope constraints. (c) Symmetrische begrenzing uitgaande van slope constraints.

Voordeel van de symmetrische begrenzing is dat de toegestane afwijking in het midden groter is dan aan het begin en eind. Grote afwijkingen in het midden kunnen immers ontstaan door optelling van vele kleine afwijkingen, terwijl aan het begin en eind alleen als gevolg van vele horizontale of verticale stappen grote afwijkingen kunnen ontstaan. Ook hier is enige voorzichtigheid met betrekking tot het uitsluiten van grote lokale tijdasformaties geboden. Begin en eindpunt van een ingesproken zin zullen in het algemeen voorafgegaan, respectievelijk gevolgd worden door een stuk stilte. De lengte hiervan hoeft niet steeds gelijk te zijn. Het lijkt daarom zinvol om aan begin en eind enkele horizontale of verticale stappen toe te staan door een onzekerheidsmarge toe te voegen. Zo komen we uiteindelijk tot de vorm die is weergegeven in figuur 2.7.



Figuur 2.7: De vorm van het toegepaste window. Linksonder en rechtsboven is ruimte gelaten voor mogelijke verschillen in de lengten van de pauzes aan begin en eind van de uitingen. De symmetrische toepassing van slopeconstraints leidt tot een ruitvormig gebied waarbinnen de warpingfunctie zich dient te bevinden. In het midden is zodoende een grotere afwijking van de diagonaal toegestaan.

De grootten van de onzekerheidsmarges kunnen door de gebruiker gespecificeerd worden. Ook indien er geen slope constraints worden toegepast blijft het window in bovenstaande vorm bestaan. In dit geval wordt hetzelfde window toegepast als in het geval $p = 1$, d.w.z. de ruit wordt begrensd door rechten met een richtingscoëfficiënt $\frac{1}{2}$ respectievelijk 2.

2.4.3 Penalties

Sakoe en Chiba [1] argumenteren dat paden in de buurt van de diagonaal bevoorreed zijn ten opzichte van andere paden omdat het aantal te passeren punten en daarmee het aantal te sommeren afstanden toeneemt bij afwijking van de diagonaal. Een juistere formulering voor de globale afstand kan verkregen worden door een normering toe te voegen:

$$G(A, B) = \frac{\sum_{k=1}^K d(c(k))}{K} \quad (2.8)$$

In deze vorm is het minimaliseringsprobleem echter niet meer met behulp van de Dynamic Programming techniek op te lossen. De waarde van K zou dan immers veranderen tijdens het minimaliseren van de subpaden, zodat op verschillende subpaden een verschillende grootte zou worden geminimaliseerd. Door een gewogen sommatie toe te passen waarbij de normering voor alle mogelijke paden dezelfde waarde N heeft, d.w.z. onafhankelijk van variaties in de warpingfunctie is, kan dit probleem worden omzeild:

$$G(A, B) = \frac{\sum_{k=1}^K w(k)d(c(k))}{\sum_{k=1}^K w(k)}, \quad (2.9)$$

met

$$\sum_{k=1}^K w(k) = N \quad (2.10)$$

De oorspronkelijke normering (delen door het aantal punten) is hiermee vervangen door een lokale normering (weegcoëfficiënten), die tijdens het oplossen van de Dynamic Programming vergelijking geëvalueerd kan worden. Het zal duidelijk zijn dat beide normeringsvormen niet identiek zijn, en niet noodzakelijk tot dezelfde warpingfunctie leiden. Er zijn voor de weegcoëfficiënten twee definities mogelijk die deze vereenvoudiging toelaten:

$w(k) = (i(k) - i(k-1)) + (j(k) - j(k-1))$, zodat $N = I + J$ (symmetrische vorm) en $w(k) = i(k) - i(k-1)$, zodat $N = I$ (asymmetrische vorm).

Een alternatieve formulering die veel gehanteerd wordt is om de weegcoëfficiënten als hellingafhankelijke strafweging (penalties) te beschouwen. Noemen we de strafweging voor een stap in horizontale richting w_1 , in diagonale richting w_2 en in verticale richting w_3 , dan vinden we voor de symmetrische en de asymmetrische vorm respectievelijk $(w_1, w_2, w_3) = (1, 2, 1)$ en $(1, 1, 0)$. Experimenteel vinden Sakoe en Chiba de beste herkenningsscore voor de symmetrische vorm. Asymmetrische wegen wijzen zij van de hand, zowel op theoretische als experimentele gronden. Ney [3] argumenteert daarentegen dat theoretisch de weging $(2, 1, \frac{3}{2})$ de voorkeur verdient. Al met al bestaat er weinig concensus over de precieze vorm van de strafweging, al is men het er wel over eens dat de toevoeging ervan noodzakelijk is om een optimale herkenningsscore te bereiken. Opvallend is dat door geen van de genoemde auteurs experimenteel de prestaties van de ongewogen versie is nagegaan.

2.5 Keuze van de norm

2.5.1 Algemeen

Voor het kwantificeren van het verschil tussen twee vectoren die een stukje spraak representeren dient een geschikte afstandsmaat gekozen te worden. Ook dit is binnen het spraakonderzoek een bekend probleem. In de loop der jaren zijn door verschillende auteurs voorstellen gedaan over de mathematische vorm van een afstandsmaat die nadruk legt op het perceptieve verschil [2, 4–9]. Pols[4] geeft een overzicht van de problematiek. Nocerino et al.[5] en Rabiner & Soong[8] hebben de prestaties van verschillende afstandsmaten in spraakherkenningsexperimenten onderzocht. De in deze experimenten onderzochte afstandsmaten zijn samengevat in tabel 2.1.

Tabel 2.1: *verschillende onderzochte afstandsmaten*

<i>afstandsmaat</i>	<i>notatie</i>
Likelihood Ratio	d_{LR}
Log Likelihood Ratio	d_{LLR}
Itakura-Saito	d_{IS}
Cepstrum	d_{Cep}^N
Bandpass Liftered Cepstrum	d_{BPCep}^N

De spraakherkenningsexperimenten gaven slechts geringe prestatieverschillen tussen de verschillende normen te zien. Bovendien hebben Gray & Markel [7] theoretisch aangetoond dat een aantal normen in de limiet $\|\vec{a} - \vec{b}\| \rightarrow 0$ naar de kwadratische norm convergeert. Voor een uitgebreidere discussie zij verwezen naar bovengenoemde literatuur. Wij zullen slechts de gekozen cepstrale afstandsmaat d_{cep}^N bespreken en onze keuze aan de hand van de eigenschappen van deze norm motiveren.

2.5.2 De cepstrale norm

Het cepstrum c_k van een discrete tijdreeks $x(n)$ is de inverse Fourier getransformeerde van de logaritme van het vermogenspectrum [5]:

$$c_k = \int_{-\pi}^{\pi} \log |X(\lambda)|^2 e^{jk\lambda} \frac{d\lambda}{2\pi} \quad (2.11)$$

Hernieuwd Fourier transformeren geeft:

$$\log |X(\lambda)|^2 = \sum_{k=-\infty}^{\infty} c_k e^{-jk\lambda} = c_0 + 2 \sum_{k=1}^{\infty} c_k \cos(k\lambda) \quad (2.12)$$

De cepstrale coëfficiënten kunnen dus ook worden opgevat als de Fouriercoëfficiënten van de logaritme van het vermogenspectrum. Een maat voor het verschil in de logaritmische vermogensspectra van twee discrete tijdreeksen $x(n)$ en $x'(n)$ wordt verkregen door een Euclidische norm toe te passen:

$$d_{L_2} = \int_{-\pi}^{\pi} |\log |X(\lambda)|^2 - \log |X'(\lambda)|^2|^2 \frac{d\lambda}{2\pi} \quad (2.13)$$

Dit kan rechtstreeks in de cepstrale coëfficiënten worden uitgedrukt:

$$\begin{aligned} d_{L_2} &= \int_{-\pi}^{\pi} \left| \sum_{l=-\infty}^{\infty} c_l e^{-jl\lambda} - \sum_{l=-\infty}^{\infty} c'_l e^{-jl\lambda} \right|^2 \frac{d\lambda}{2\pi} = \\ &= \sum_{l=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} (c_l - c'_l)(c_m - c'_m) \int_{-\pi}^{\pi} e^{-j(l-m)\lambda} \frac{d\lambda}{2\pi} = \\ &= \sum_{l=-\infty}^{\infty} (c_l - c'_l)^2 \end{aligned} \quad (2.14)$$

De cepstrale norm wordt verkregen door de reeks na N termen af te breken:

$$d_{cep}^N(x, x') = \sum_{l=-N}^N (c_l - c'_l)^2 \quad (2.15)$$

Aangezien $c_k = c_{-k}$ kan dit nog herschreven worden als

$$d_{cep}^N(x, x') = c_0 + 2 \sum_{l=1}^N (c_l - c'_l)^2 \quad (2.16)$$

Later zal blijken dat c_0 een bijzondere rol speelt. Het afbreken van de reeks houdt in dat de logaritmische vermogensspectra worden benaderd met de eerste N termen van hun Fourierreeksen. Dit heeft daarom een smoothing en een Gibbs effect tot gevolg.

De cepstrale coëfficiënten kunnen recursief berekend worden uit de gain en de LPC coëfficiënten, $[\sigma, a_1, \dots, a_p]$ en vice versa [5]. De recursie relatie luidt:

$$\begin{aligned} c_0 &= \ln \sigma^2 \\ c_k &= -a_k - \sum_{i=1}^{k-1} a_i c_{k-i} \frac{k-i}{k}, \quad 1 \leq k \leq p \\ c_k &= -\sum_{i=1}^p a_i c_{k-i} \frac{k-i}{k}, \quad k > p \end{aligned} \quad (2.17)$$

De eerste coëfficiënt is alleen afhankelijk van de gain, en is niet nodig om de volgende coëfficiënten te bepalen. De gain kan daarom naar believen worden gekozen. Deze keuzevrijheid is noodzakelijk daar luidheidsverschillen ten gevolge van veranderingen in opname-omstandigheden of verschillen in sprekerluidheid niet in de afstandsmaat mogen doorwerken. Normalisatie van de energie op de waarde 1 ($c_0 = 0$) elimineert de luidheid volledig uit de afstandsmaat. Naast deze keuze hebben Gray & Markel [7] een aantal plausible voorstellen gedaan omtrent de keuze van σ . Om het onderscheid tussen stemloze en stemhebbende klanken te benadrukken kan een niet-volledige eliminatie van de gain de voorkeur genieten.

"energieverschillen"
moet zijn "energie"

Gray & Markel [7] hebben aangetoond dat het afbreken van de reeks voor $N >$ orde van de LPC een goede benadering geeft voor de in (2.13) gedefinieerde Euclidische afstand in het logaritmisch spectrum d_{L_2} [5]. Voor 10^e orde LPC analyse vonden zij tussen d_{cep}^{10} en d_{L_2} een correlatie van 0.98.

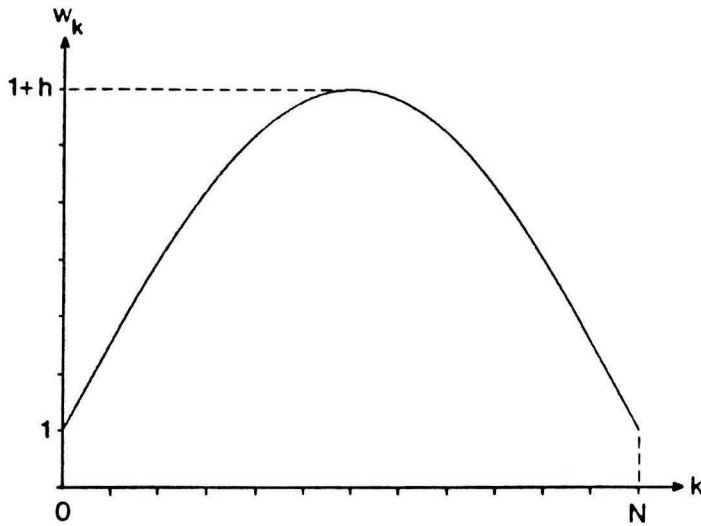
2.5.3 Bandpass liftered cepstrale afstand

Tohkura [8] laat in een experiment zien dat de van LPC afgeleide cepstrale coëfficiënten onderling grote verschillen in variantie vertonen, d.w.z. hetzelfde woord gesproken door verschillende sprekers leidt bij de eerste 3 à 4 coëfficiënten tot een veel grotere spreiding dan bij de hogere coëfficiënten. In een woordherkenningsexperiment waarbij een weging in de afstandsmaat de lagere coëfficiënten onderdrukt, vindt hij een substantiele verbetering van

de herkenningsscore. Dit alles suggereert dat vooral spreker, spreekwijze en transmissiekanaal afhankelijke verschillen leiden tot een grote spreiding in de lagere cepstrale coëfficiënten. Tenslotte vindt hij dat de herkenningsscore afneemt indien de hoogste coëfficiënten te zwaar worden gewogen. Hierbij gaat het vooral om de coëfficiënten hoger dan de LPC-orde. Mogelijke oplossingen hiervoor zijn het tijdig afbreken van het cepstrum of het toepassen van een symmetrische weging. Dit laatste komt neer op een bandpass liftering. Ook Juang, Rabiner & Wilpon [9] komen tot de conclusie dat bandpass liftering noodzakelijk is om ongewenste varianties in de via LPC bepaalde cepstrale coëfficiënten te onderdrukken. Zij vinden experimenteel voor de hogere coëfficiënten grotere varianties dan verwacht volgens een statistisch model en argumenteren dat dit een inherent artefact van het LPC spectrum is. Zij stellen een bandpass liftering voor volgens een sinusboog:

$$w_k = 1 + h \sin\left(\frac{k\pi}{N}\right) \quad (2.18)$$

Hierin bepaalt h de vorm van de lifter. N bepaalt de breedte en is gelijk aan het aantal benodigde cepstrale coëfficiënten. In een herkenningsexperiment vinden zij uitgaande van 8^e orde LPC analyse optimale waarden $h = 6$ en $N = 12$. De vorm van de bandpasslifter is in figuur 2.4 weergegeven.



Figuur 2.8: Vorm van de Bandpasslifter w_k .

Rabiner & Soong [6] nemen in hun vergelijkende studie van verschillende normen voor de bandpass liftered cepstrale norm $N = 12$ bij $f_s = 6.6$ kHz en 10^6 orde LPC. We hebben reeds eerder gezien dat het aantal cepstrale coëfficiënten evenredig is met het detail in het logaritmisch vermogenspectrum. Verandering van de samplefrequentie leidt eveneens tot evenredige verandering in detail van het spectrum. Het lijkt daarom aannemelijk het aantal cepstrale coëfficiënten aan te passen aan de samplefrequentie. Uitgaande van $N = 12$ bij $f_s = 6.6$ kHz nemen we daarom in onze toepassing ($f_s = 10$ kHz) $N = 12 * 10/6.6 = 18$ als uitgangspunt. De hoogte h van de bandpass lifter wordt hierbij niet veranderd.

Hoofdstuk 3

Implementatie

3.1 Korte beschrijving

Het geïmplementeerde programma maakt een analyse van twee N-files en schrijft het resultaat weg naar twee A/P-files en een DTW-file. De A/P-files zijn alleen van belang indien men achteraf geïnteresseerd is in de analyseparameters. De voor de warping benodigde analysegegevens worden via de DTW-file doorgegeven aan het warpingalgoritme. Uit de tijdens de analyse bepaalde reflectiecoëfficiënten worden de cepstrale coëfficiënten berekend. Vervolgens wordt een warping bepaald. Het resultaat wordt weggeschreven in de vorm van een text-file en een figuur. Hierop komen we in paragraaf 3 uitgebreider terug.

3.2 De interactie met de gebruiker

Na aanroep verschijnt eerst een aantal vragen met betrekking tot de files:

input A/P-files:

input N-files:

output A/P files:

De eerste vraag kan met < cr > beantwoord worden. Het is voor het

bepalen van een warping tussen twee uitingen voldoende om de betreffende N-files te specificeren.

De input N-files dienen met opeenvolgende nummering te worden opgegeven, bijvoorbeeld *test1*, *test2*. Twee verschillende namen ingeven is niet mogelijk. Indien alleen een naam wordt gegeven, zonder nummer, wordt daarna naar begin- en eindnummer gevraagd. De eerste N-file (in ons voorbeeld *test1*) wordt in het vervolg aangeduid met de term 'input', de tweede met 'template' of 'reference'.

De output A/P-files die het resultaat van de analyse bevatten zijn niet van belang indien de gebruiker alleen geïnteresseerd is in een warping. De voor de warping benodigde gegevens worden tijdens de analyse immers naar een DTW-file geschreven. Indien de vraag naar de output A/P filenaam met < cr > wordt beantwoord, wordt aan deze files dezelfde naam gegeven als de bijbehorende N-files.

Hierna verschijnt een aantal vragen omtrent de analyseparameters:

LPC analysis order (cr=12):

Frame length (cr=25(ms)):

Frame hopping (cr=10(ms)):

Preemphasis factor (cr=0.9):

Tenslotte volgt een aantal vragen met betrekking tot de Dynamic Time Warping:

Slope constraint p ($cr=0$)

(0: no constraints, 0.5: $y=3x$ and $y=1/3 x$, 1: $y=2x$ and $y=1/2 x$):

Penalties horizontal, diagonal, vertical ($cr=(1,1,1)$):

Uncertainties begin of input, begin of template ($cr=(10, 10)$):

Uncertainties end of input, end of template ($cr=(10, 10)$):

Width L of bandpass lifter ($cr=18$):

Height h of bandpass lifter ($cr=6$):

De parameter p is een maat voor de maximaal toegestane hellingen van de warping functie. De richtingscoëfficiënt van de warpingfunctie is beperkt tussen de waarden $\frac{p}{1+p}$ en $\frac{1+p}{p}$. Er kan gekozen worden uit $p = 0$, $p = \frac{1}{2}$ of $p = 1$. Toepassen van slope constraints ($p \neq 0$ kiezen) wordt bij lange spraakuitingen afgeraden, omdat bijvoorbeeld grote verschillen in rustpauzes dan niet goed kunnen worden gewarped.

Penalties kunnen worden toegepast om te voorkomen dat paden in de buurt van de diagonaal bevoordeeld worden omdat zij minder stappen nodig hebben dan andere paden. De keuze (1,2,1) elimineert de invloed van het aantal stappen volledig uit de bepaling van het pad. Het effect van slope constraints en penalties op de warpingfunctie, kan worden nagegaan aan de hand van de in het volgende hoofdstuk beschreven tests.

De onzekerheidsmarges (uncertainties) dienen om ervoor te zorgen dat verschillen in pauzelengte voor en na de spraakuitingen effectief gewarped kunnen worden. Binnen deze marges worden geen beperkingen aan het pad opgelegd (d.w.z. geen window en $p = 0$). 'Input' slaat hier op de als eerste ingevoerde N-file (N-file met het laagste volgnummer), 'template' op de tweede. De instelling van de uncertainties is niet kritisch. Alleen te kleine marges kunnen een slechte warping tot gevolg hebben. De defaultwaarden zullen in de meeste gevallen voldoende ruimte geven.

De breedte van de bandpasslifter dient afhankelijk van de samplefrequentie f_s te worden gekozen. Uitgegaan is van $f_s = 10$ kHz. Het verband tussen de breedte van de lifter en de samplefrequentie is lineair verondersteld: $N = \frac{f_s}{10^3} 18$.

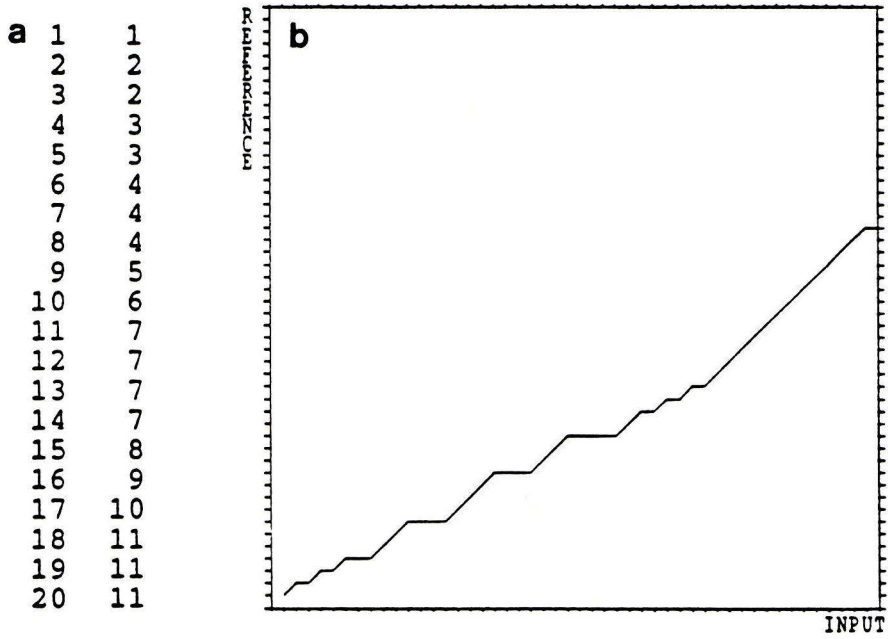
De hoogte h van de lifter is theoretisch niet van andere parameters afhankelijk. Er bestaat geen reden aan te nemen dat de defaultwaarde aanpassing behoeft.

In het algemeen zijn de defaultwaarden van de DTW parameters zodanig gekozen dat een redelijke warping verwacht kan worden. Hierbij is niet altijd gekozen voor de uit woordherkenningsonderzoek gebleken optimale waarden, maar heeft met het oog op warping van langere uitingen een soepelere instelling de voorkeur gekregen (slopeconstraints, penalties). Indien een onmogelijke waarde wordt opgegeven voor een van de parameters, zal hiervan een melding op het scherm verschijnen en zullen de vragen worden herhaald.

3.3 Uitvoer

De uitvoer bestaat uit een figuur en een text-file (zie figuur 3.1). De figuur is aangepast om op de laserprinter als vierkant te verschijnen (via de gebruikelijke LVS procedure 'setterm laser'). Op het scherm zal het een enigszins rechthoekige vorm krijgen.

Indien lange uitingen gewarped worden, is aan de hand van een figuur vaak moeilijk te zien welke frames precies op elkaar afgebeeld worden. In dat geval kan de text-file met de volgens de warping bij elkaar horende framenummers worden geraadpleegd. Deze bestaat uit twee kolommen. De linker kolom bevat de framenummers van de input (de als eerste ingevoerde uiting) en de rechterkolom de bijbehorende framenummers van de reference (de als tweede ingevoerde uiting). De text-file wordt weggeschreven onder de naam 'Path-inp-ref'.



Figuur 3.1: Voorbeeld van de programma-uitvoer. (a) De eerste paar regels van een text-file en (b) de bijbehorende figuur.

Hoofdstuk 4

Testen van het programma

Bij het zoeken naar geschikte testdata deed zich het probleem voor dat handmatige controle van de juistheid van het gevonden pad in het gegeven tijdsbestek ondoenlijk was. Bovendien stond geen meervoudige realisatie van eenzelfde zin door de zelfde spreker tot onze beschikking, zodat testen van het programma in een situatie gelijkend op die waarvoor het in eerste instantie bedoeld is niet mogelijk was. Om toch een indruk te krijgen van de werking van het programma en de invloed van diverse parameterkeuzen zijn de volgende testen uitgevoerd:

- A. *Warping van een zinsdeel na kunstmatige tijdasmanipulatie tegen de oorspronkelijke versie.*
- B. *Warping van realisaties van eenzelfde zin door verschillende sprekers.*

De kunstmatige tijdasmanipulaties zijn uitgevoerd met behulp van het programma MAD (MANipulation of Duration)¹, en bestonden uit het vermenigvuldigen van een deel van de tijdas met een constante factor. Van een realisatie van de zin 'Maandag gaan we naar het zwembad' door een mannelijke spreker werd het gedeelte 'maanda' van het eerste woord in tijdsduur verdubbeld. Van het eerste woord 'maandag' werd een warping tussen de gemanipuleerde en de oorspronkelijke versie bepaald. In tabel 4.1 zijn de default instellingen van het programma samengevat. In tabel 4.2 staan de in de tests gebruikte realisaties van zinnen of zinsdelen met

¹Ir J.H. Eggen, kamer 2.20

hun bijbehorende code. De code bestaat uit een identificatie van de zin (meestal een gedeelte van een woord uit de zin) met een toevoeging M of V (sekse van de spreker) en een nummer van de spreker. In tabel 4.3 zijn de relevante gegevens van de testdata samengevat. Parameterwaarden zijn alleen aangegeven indien ze afwijken van de default instellingen uit tabel 4.1. In de kolommen *input* en *reference* staan de zincodes van de gewarpte uitingen.

We zullen niet alle testresultaten uitvoerig beschrijven, maar slechts aan de hand van een bloemlezing de belangrijkste bevindingen illustreren. De machine-uitvoer van de testen is in appendix A opgenomen.

Tabel 4.1: *Default instellingen*

<i>parameter</i>	<i>waarde</i>
LPC analysis order	12
Frame length	25 ms
Frame hoping	10 ms
Preemphasis factor	0.9
p (slope constraints)	0
Penalties	(1,1,1)
Uncertainties begin	(10,10)
Uncertainties end	(10,10)
L (width of BPL)	18
h (height of BPL)	6

Tabel 4.2: *De zincodes*

<i>code</i>	<i>zin(sdeel)</i>
rijdXX	Rijden onder invloed is strafbaar.
voorXX	Toch lijkt me dat een goed voorstel.
maandag	maandag
maandaghalf	maandag

Tabel 4.3: Gegevens van de testdata

<i>test</i>	<i>input</i>	<i>reference</i>	<i>parameterwaarden</i>
A1	maandag	maandaghalf	
A2	maandag	maandaghalf	p=0.5
A3	maandag	maandaghalf	p=1
A4	maandag	maandaghalf	p=0.5, penalties (1,2,1)
A5	maandag	maandaghalf	p=1, penalties (1,2,1)
B1	rijdm1	rijdm2	
B2	rijdm2	rijdm3	
B3	rijdm1	rijdm3	
B4	rijdv3	rijdv5	
B5	rijdm1	rijdv3	
B6	rijdm2	rijdv3	
B7	rijdm3	rijdv3	
B8	rijdm1	rijdv5	
B9	rijdm2	rijdv5	uncertainties (20,20) (20,20)
B10	rijdm3	rijdv5	uncertainties (20,20) (20,20)
B11	voorm2	voorm4	uncertainties (20,20) (20,20)
B12	voorm2	voorv2	uncertainties (20,20) (20,20)
B13	voorm2	voorv5	uncertainties (20,20) (20,20)
B14	voorm4	voorv2	uncertainties (20,20) (20,20)
B15	voorm4	voorv5	

4.1 Kunstmatige tijdas manipulatie

De theoretisch verwachte warpingfunctie na tijdas manipulatie is in figuur 4.1 weergegeven. Deze bestaat uit een stuk met richtingscoëfficiënt $\frac{1}{2}$ en een stuk met richtingscoëfficiënt 1.

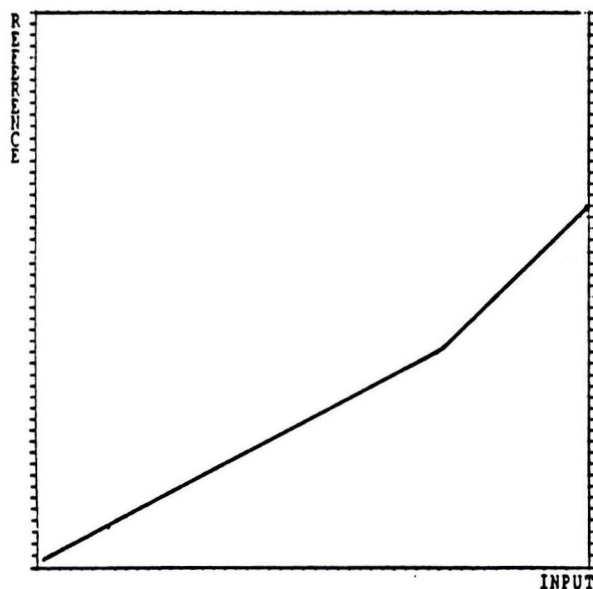
Test A1 (alle instellingen default) geeft globaal de verwachte hellingen te zien (zie figuur 4.2). Lokaal treden nog wel kleine afwijkingen op die met het oog op de wijze van de tijdasmanipulatie niet direct als foutief bestempeld kunnen worden.

Intensivering van de slope constraints ($p = 0.5$, respectievelijk 1 in tests A2 en A3) heeft een onderdrukking van deze lokale afwijkingen tot gevolg (zie figuur 4.3 en 4.4). Dit ligt in de lijn der verwachting, immers bij toenemende beperking van de maximaal toegestane hellingen is het niet

meer mogelijk snelle variaties in de warpingfunctie aan te brengen.

Toepassen van symmetrische penalties $(w_1, w_2, w_3) = (1, 2, 1)$ heeft een tegengesteld effect. Het straffen van diagonale stappen leidt zoals verwacht tot grotere lokale afwijkingen van de diagonaal (zie figuur 4.5). Bij een combinatie van slope constraints en symmetrische weging hebben de slope constraints een overheersende invloed (zie figuur 4.6). Ook dit is in overeenstemming met de verwachtingen. Slope constraints verbieden immers snelle variaties ten opzichte van de diagonaal, terwijl penalties dergelijke variaties slechts aanmoedigen.

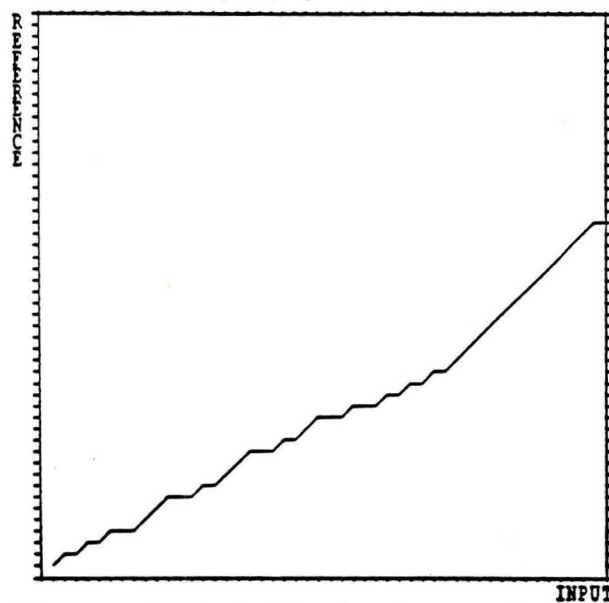
Daar begin en eindpunt van de oorspronkelijke en de tijdsduurgemanipuleerde uiting goed overeen komen, d.w.z. geen groot verschil in pauzes vooraf en achteraf, hebben veranderingen in de onzekerheidsmarges van het window geen invloed op het pad.



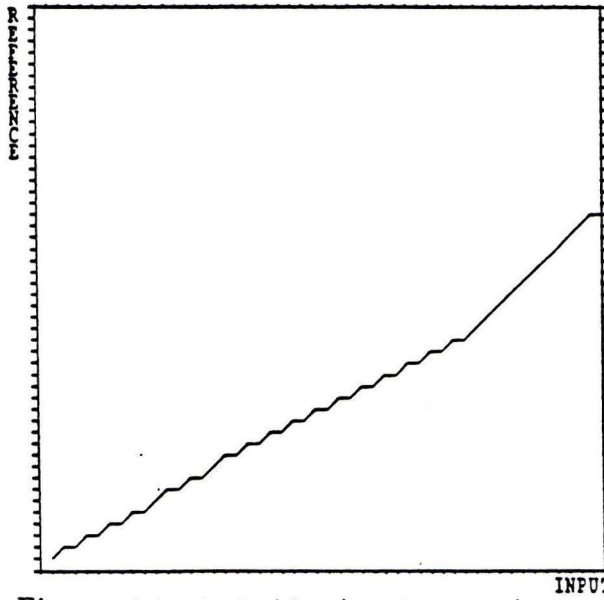
Figuur 4.1: *De theoretisch verwachte warpingfunctie na kunstmatige tijds manipulatie (testserie A). Input=maandag, reference=maandaghalf. Het eerste gedeelte heeft een richtingscoëfficiënt $1/2$, het tweede gedeelte een richtingscoëfficiënt 1 .*



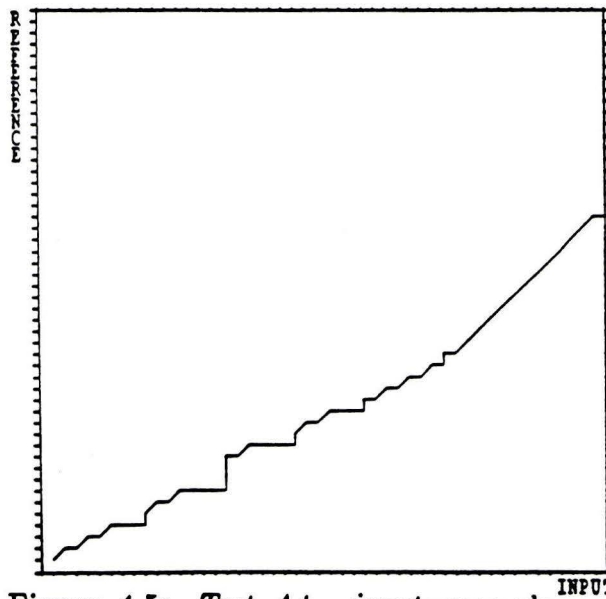
Figuur 4.2: *Test A1. input=maandag, refer-
ence= maandaghalf, p=0.*



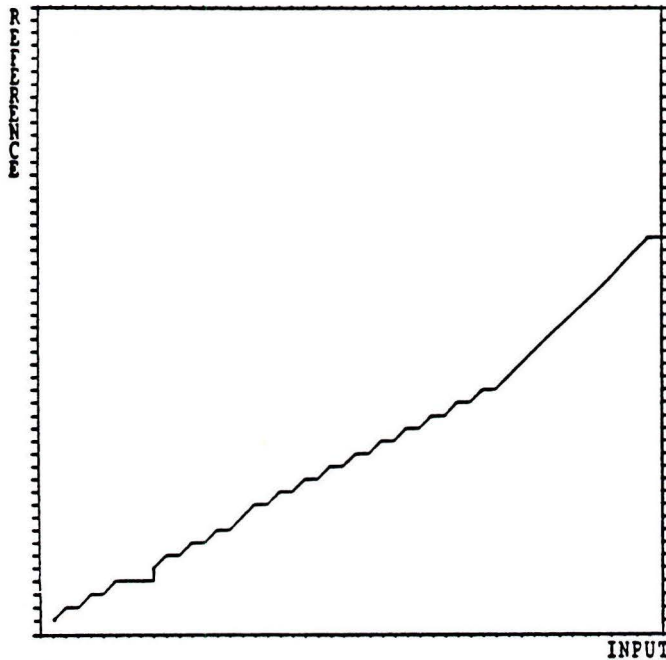
Figuur 4.3: *test A2. input=maandag, refer-
ence= maandaghalf, p=0.5.*



Figuur 4.4: *test A3*. *input=maandag*, *reference= maandaghalf*, $p=1$.



Figuur 4.5: *Test A4*. *input=maandag*, *reference= maandaghalf*, *symmetrische penalties (1,2,1)*, $p=0$.



Figuur 4.6: *Test A5. input=maandag, reference= maandaghalf, symmetrische penalties (1,2,1), $p=1$.*

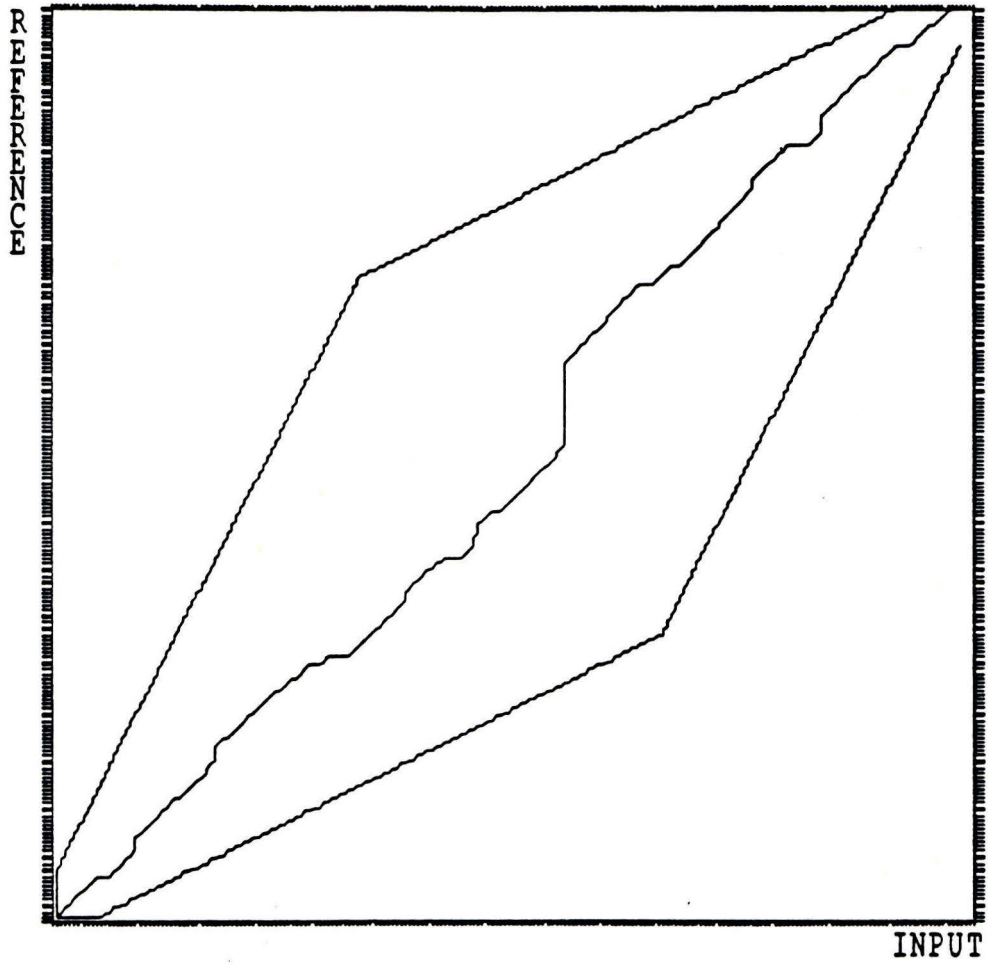
4.2 Realisaties van eenzelfde zin door verschillende sprekers

Test B2 (rijdm2/rijdm3) geeft een opvallend lang vertikaal stuk in het pad te zien (zie figuur 4.7). Dit nodigde uit om beide zinnen nogmaals te beluisteren. Spreker m3 bleek inderdaad hoorbaar een korte rustpauze te hebben ingelast. Merk op dat het optreden van dergelijke pauzes onder toepassing van slope constraints niet goed zou kunnen worden weergegeven door de warpingfunctie.

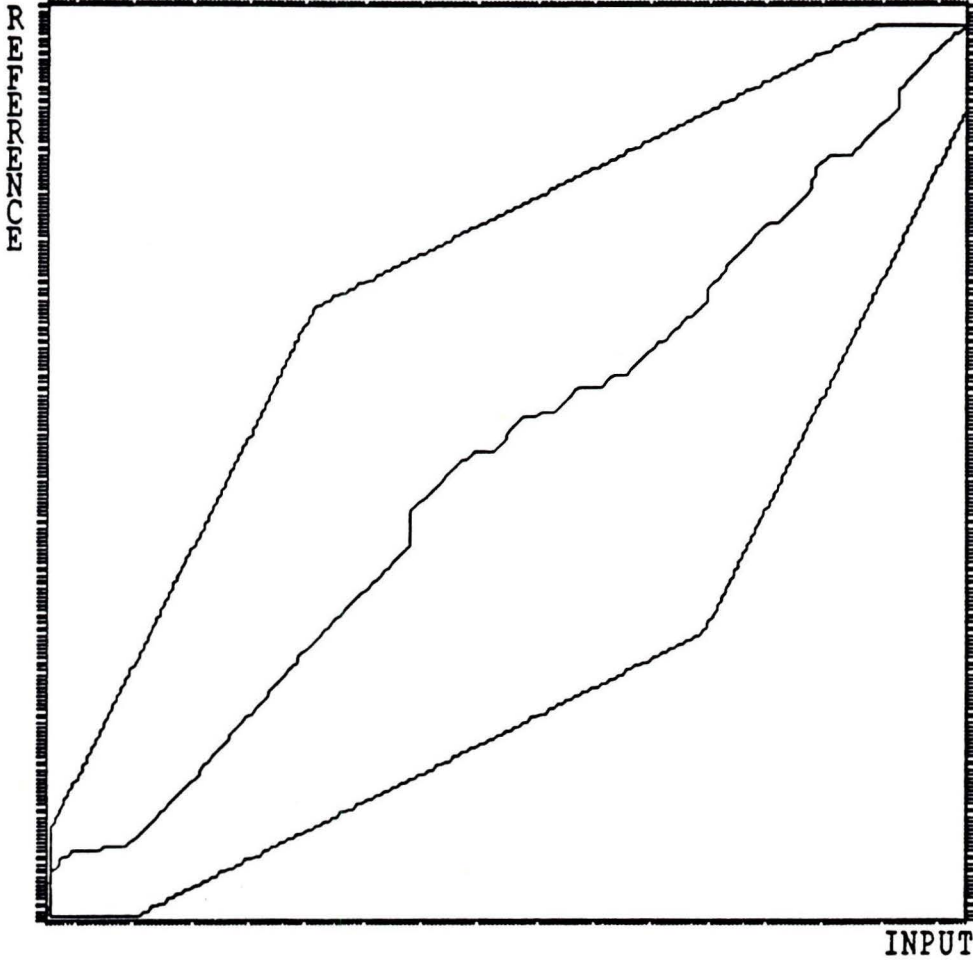
Test B11 (voorm2/voorm4) toont de noodzakelijkheid van de onzekerheidsmarges in het window aan (zie figuur 4.8). Een van beide realisaties werd voorafgegaan door een langere pauze.

In het algemeen is het moeilijk na te gaan of de warpingfuncties uit testserie B goede weergaven zijn van de tijdasdeformaties tussen de verschil-

lende uitingen. Handmatige controle was binnen het gegeven tijdsbestek niet mogelijk.



Figuur 4.7: *Test B2. input=rijdm2, reference=rijdm3, alle instellingen default.*



Figuur 4.8: *Test B11. input=voorm2, reference=voorm4, alle instellingen default.*

4.3 Conclusies

Het programma blijkt geen evidente fouten in de uitvoering van het algoritme en de berekeningen te bevatten.

Slope constraints maken het onmogelijk grote deformaties ten gevolge van bijvoorbeeld rustpauzes in realisaties van zinnen correct weer te geven. Eventuele toepassing van slope constraints dient daarom eerst zorgvuldig overwogen te worden.

Strengere penalties voor stappen in diagonale richting leiden tot grote tijdasdeformaties in opeenvolgend tegengestelde richting, en hebben vrijwel geen effect indien gelijktijdig slope constraints worden toegepast. Het toepassen van penalties lijkt daarom nauwelijks tot een verbetering van de warpingfunctie te leiden.

Literatuur

- [1] Sakoe, H. & Chiba, S. (1978) , Dynamic programming algorithm optimization for spoken word recognition, *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-26 pp. 43-49.
- [2] Itakura, F. (1975) , Minimum prediction residual principle applied to speech recognition, *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-23 pp. 67-72.
- [3] Ney, H. (1984) , The use of a one-stage dynamic programming algorithm for connected word recognition, *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-32 pp. 263-271.
- [4] Pols, L.C.W (1987), Distance measures; physical and perceptual aspects, *Colloquium Signaalanalyse & Spraak*, 23-25 sept. 1987, Amsterdam.
- [5] Nocerino, N., Soong, F.K., Rabiner, L.R. & Klatt, D.H. (1985) , Comparative study of several distortion measures for speech recognition, *Speech Communication* 4, 317-331.
- [6] Rabiner, L.R & Soong, F.K. (1985) , Single-frame vowel recognition using vector quantization with several distance measures, *AT&T Technical Journal*, Vol. 64, No. 10, pp. 2319-2330.
- [7] Gray, A.H. & Markel, J.D. (1976) , Distance measures for speech processing, *IEEE Trans. Acoust., Speech, Signal Processing*, Vol 24, pp. 380-391.
- [8] Tohkura, Y. (1986) , A weighted cepstral distance measure for speech recognition, *IEEE, ICASSP 86, Tokyo*, 14.17, pp. 761-764.
- [9] Juang, B.H., Rabiner, L.R. & Wilpon, J.G. (1986) , On the use of bandpass filtering in speech recognition, *IEEE, ICASSP 86, Tokyo*, 14.18, pp. 765-768.

Appendix A

Machine-uitvoer van de programmatests

