ELSEVIER

Image and Vision Computing 20 (2002) 597–607

image AND vision COMPUTING

www.elsevier.com/locate/imavis

A non-linear model of shape and motion for tracking finger spelt American sign language

Richard Bowden[a,*], Mansoor Sarhadi[b]

[a]*CVSSP, School of ECM, University of Surrey, Guildford, Surrey GU2 7XH, UK*
[b]*Department of Systems Engineering, Brunel University, Uxbridge, Middlesex UB8 3PH, UK*

Received 10 June 2001; received in revised form 19 February 2002; accepted 14 March 2002

Abstract

This work presents a piecewise linear approximation to non-linear Point Distribution Models for modelling the human hand. The work utilises the natural segmentation of shape space, inherent to the technique, to apply temporal constraints, which can be used with CONDENSATION to support multiple hypotheses and discontinuous jumps within shape space. This paper presents a novel method by which the one-state transitions of the English Language are projected into shape space for tracking and model prediction using an HMM like approach. The paper demonstrates that this model of motion provides superior results to that of other tracking approaches.

Keywords: Deformable model; Hidden Markov model; CONDENSATION; Particle filtering; Eigenshapes; Gesture recognition; Sign language

1. Introduction

Sign language is the natural language of the deaf community. It is a rich and expressive language that has its own rules of grammar, structure and composition. Different geographic regions have their own sign language due to the isolation, history and requirements of that community. However, they often reflect the language of the hearing society in which they live. As a product of this, sign languages typically consist of 2 main elements:

- a simple signed alphabet which mimics the letters of the native spoken language;
- a higher level signed language which conceptualises pronouns, verbs, nouns and adjectives as meaningful gestures often using actions to mimic the meaning or description of the sign.

The purpose of the signed alphabet, termed finger-spelling, is that it allows names or words to be spelt when no sign is either known or present. It is also integrated within the sign vocabulary, e.g. in British Sign Language, tapping the letter 'M' twice is the sign for mother. Two sign languages, which have a common underlying spoken language, are American Sign Language (ASL) and British Sign Language (BSL), the major difference between the two being a one- or two-handed system, respectively. Of course, the major sign vocabulary varies immensely between them. ASL provides a more suitable problem domain over British Sign Language, as the BSL finger spelt alphabet is a two-handed system. This presents added difficulty for computer vision approaches due to the problems associated with occlusion. Fig. 1 shows the ASL alphabet with the corresponding hand pose for each letter of the alphabet.

Systems such as Simon the virtual signer [16] have been developed which allow a human avatar to convert written text into a multimedia BSL video stream. However, to-date no system exists which has sufficient reliability and vocabulary to convert sign to speech at the level required for translation. It is obvious that any system that can visually interpret sign would provide a significant tool for deaf-hearing communication.

Gesture recognition in computer vision is an extensive area of research that encompasses anything from static pose estimation of hands or body to dynamic movements such as a wave or a shoulder shrug. Such an extensive review is beyond the scope of this paper and the interested reading is directed to a number of both online[1] and published surveys

* Corresponding author.
E-mail address: r.bowden@surrey.ac.uk (R. Bowden).

[1] The gesture recognition homepage—www.cybernet.com/~ccohen.pdf; a brief overview of gesture recognition—www.dai.ed.ac.uk/Cvonline/LOCAL_COPIES/COHEN/gesture_overview.html; vision based hand gesture recognition systems—ls7—www.cs.uni-dortmund.de/research/gesture/vbgr-table.html.

0262-8856/02/$ - see front matter
PII: S0262-8856(02)00049-5

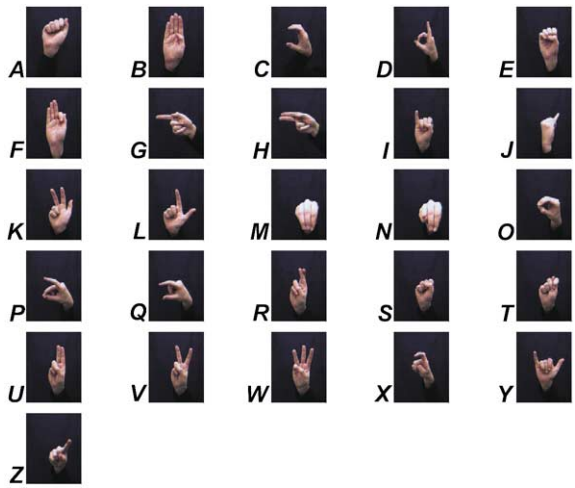

Fig. 1. The American sign language finger spelling alphabet.

[15]. Gesture recognition systems typically involve extremely small vocabularies selected for easily distinguishable characteristics. Sign language recognition, by default, requires a large vocabulary proving a more difficult research task and hence has attracted less attention.

Few authors have attempted system using the major vocabulary of sign. Starner and Pentland [11] developed a system capable of recognising a subset of 40 ASL signs using a Hidden Markov Model (HMM) and a course representation of hand shape, orientation and trajectory. Vogler and Metaxas [10] extended tracking to 3D for a vocabulary of 53 signs again using an HMM for recognition. Other authors [12–14,17] have investigated systems for finger spelt sign recognition. Gao used [12] a chain code based representation and a neural network to achieve the recognition of thirteen pre-defined hand postures. Freeman [17] describes a system to recognise 15 postures using orientation histograms and nearest neighbour classification. Recently a number of authors have proposed size functions as a method to represent the sign alphabet [13,14]. However, for finger spelling, these techniques are floored in that they only recognise static poses of the hand.

This paper is less concerned with classification, but with the robust tracking of the hand during signing. Without the ability to track robustly, classification is not possible. To enable tracking we construct a temporal model of shape and motion.

Previous work by the author and other researchers have investigated statistical models of deformation [1–8]. These deformable models have been used to learn a priori shape and deformation from a training set of examples which represent the shape and deformation of an object or a class of objects. Statistical models of deformation are typically constructed with prior knowledge about deformation but the temporal context of this deformation is ignored. This constraint is beneficial in disambiguating model pose during tracking.

A large body of work has been performed on the temporal mechanics of tracking. Many researchers have attempted to use predictive methods such as those based within a Kalman filter framework [1]. Hill et al. [6,7] proposed using genetic algorithms to model the discontinuous changes in shape space[2]/model parameters. Of particular interest to the work presented here is the CONDENSATION algorithm [1,8] (also known as particle filtering), which is a method for stochastic tracking, where a population of model hypotheses are generated at each iteration. These populations are generated from Probability Density Functions (PDFs) generated over the model parameter space to provide a hypothesis-and-test approach to model prediction and tracking.

The key to this approach is an a priori model of motion from which populations are generated. Where motion is relatively uniform, such as the motion of an object within an image, the learning stage can be bootstrapped to the tracking process [8]. However, for the movement of the model within shape space (the deformation parameter space) this is not possible [4]. This is due to the inherently high dimensionality and complex dynamics of shape spaces and the computational limitations of the CONDENSATION algorithm. Instead, motion models must be learnt in much the same way as deformable models; the temporal model augments that of deformation and provides an indication of where in shape space the object may move to next given its preceding shape. Unfortunately, as we will see in Section 4, although a relatively small sample of training data can be used to construct a model of deformation, considerably more examples are required to achieve an accurate representation of motion over the entire parameter space.

This paper addresses the problem of constructing a non-linear deformable model of the human hand for sign language recognition. It is demonstrated that the temporal model cannot be constructed from training data alone and a method which, allows temporal information about the English language to be projected into shape space is presented. This generates a first order temporal model, which incorporates both information about shape space and the English Language.

Section 2 discusses the construction of a non-linear Constrained Shape Space Point Distribution Model (CSSPDM [4]) using a piecewise linear approximation. Section 3 demonstrates how the CSSPDM naturally lends itself to a CONDENSATION like approach to tracking. Section 4 presents a method by which the first order transitions of the English Language are propagated into shape space using a Hidden Markov Model like approach. Finally, the approaches are compared and conclusions drawn.

[2] Shape space refers to the model parameter space where changes in those parameters result in a deformation of the original model, i.e. a change in shape.

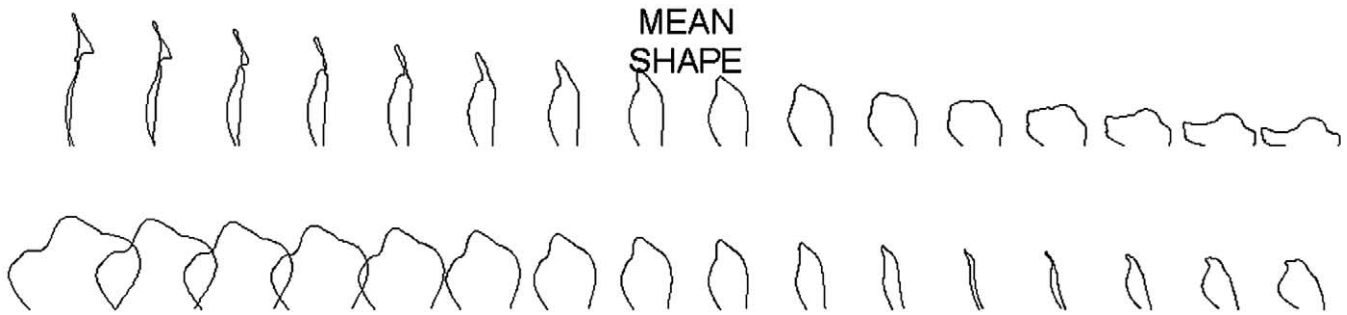

Fig. 2. First and second primary modes of the ASL model.

2. Constructing a CSSPDM for sign language

2.1. Constructing a linear Point Distribution Model

A Point Distribution Model (PDM) is generated by performing principal component analysis upon a training set to form a linear model of deformation [5]. PDMs have proven themselves an invaluable tool in image processing. The *classic formulation* combines local edge feature detection and a model-based approach to provide a fast, simple method of representing an object and how its structure can deform.

To construct a PDM a 2D contour is described by a vector $\mathbf{x}_i \in R^{2n} = (x_1, y_1, \ldots, x_n, y_n)^{\mathrm{T}}$, representing a set of n points specifying the path of the contour. A training set $\mathbf{E}$ of N vectors is then assembled for a particular model class. The training set is aligned (using translation, rotation and scaling) and the mean shape calculated. To represent the deviation within the shape of the training set, Principal Component Analysis (PCA) is performed on the deviation of the example vectors from the mean using an eigenvector decomposition on the covariance matrix $\mathbf{S}$ of $\mathbf{E}$ [5], where

$$\mathbf{S} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^{\mathrm{T}}, \qquad \bar{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

PCA projects the data into a linear subspace with a minimum loss of information by multiplying the data by the eigenvectors of the covariance matrix ($\mathbf{S}$). By analysing the magnitude of the corresponding eigenvalues the minimum dimensionality of the space on which the data lies can be calculated and the information loss estimated [2].

The t unit eigenvectors of $\mathbf{S}$ corresponding to the t largest eigenvalues supply the variation modes; t will generally be much smaller than N, thus giving a very compact model and it is this dimensional reduction that will facilitate non-linear analysis. A deformed shape $\mathbf{x}$ is generated by adding weighted combinations of v_j to the mean shape,

$$\mathbf{x} = \bar{\mathbf{x}} + \sum_{j=1}^{t} b_j \mathbf{v}_j$$

where b_j is the weighting for the jth variation vector. Suitable limits for b_j are $\pm 3\sqrt{\lambda_j}$, where λ_j is the jth largest eigenvalue of $\mathbf{S}$ [13] and $\sqrt{\lambda_j} \equiv \sigma_j$, the standard deviation of the distribution along the eigenvector. This provides a compact mathematical model of how the shape deforms.

The formulation of the PDM can also be expressed in matrix form [5]:

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}$$

where $\mathbf{P} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_t)^{\mathrm{T}}$ is a matrix of the first t eigenvectors and $\mathbf{b} = (b_1, b_2, \ldots, b_t)^{\mathrm{T}}$ is a vector of weights.

This mathematical model is used to constrain the shape of the PDM when applied to an image. To locate and track an object the Active Shape Model algorithm [5] is used. A contour is placed near to the desired feature in the image plane. The fitting process is an iterative one, whereby the contour makes small steps within the image to find a natural resting-place and is effectively a gradient descent method. The model uses suggested movements from control points (using edge detection or grey level matching). Movement of the model is then allowed through the relocation of the model within the image plane using translation, rotation, and scaling. Deformation of the model is also permitted by finding the closest allowable shape as determined by the bounds of the mathematical model of deformation. Given a new shape $\mathbf{x}'$, the closest allowable shape from the model is constructed by finding $\mathbf{b}'$ such that

$$\mathbf{b}' = \mathbf{P}^{-1}(\mathbf{x}' - \bar{\mathbf{x}}), \qquad -3\sqrt{\lambda_i} \le b'_i \le 3\sqrt{\lambda_i}$$

The closest allowable shape can then be reconstructed as

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}'$$

2.2. The linear ASL PDM

Several image sequences were recorded of a subject signing. These consisted of numerous occurrences of each of the letters of the alphabet. The sequences included three 'runs' through the alphabet, along with a small selection of simple sentences and words. Once these sequences had been extracted, the hand was segmented to produce a binary image, and a contour-tracing algorithm initiated to extract the external contour of the hand for each image frame. After standard alignment and resampling of the contour to 200

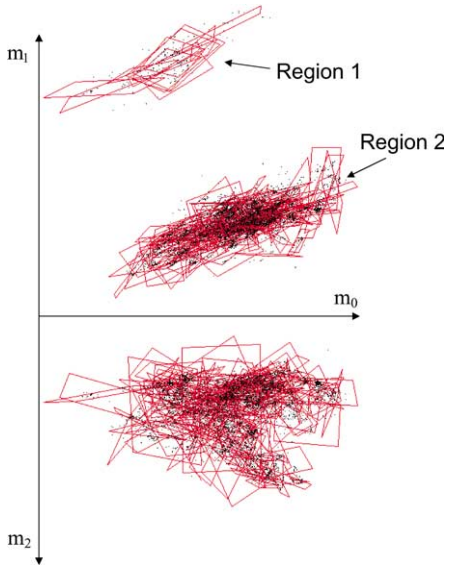

Fig. 3. Visualisation of constrains applied within shape space for the ASL CSSPDM.

points (as described in Ref. [4]) a training set of 7441 examples (approximately 250 per signed letter) was produced where each pose is described by a vector $\mathbf{x}_i \in R^{400} = (x_1, y_1, \ldots, x_{200}, y_{200})^{\mathrm{T}}$. Once assembled the procedure outlined in Section 2.1 is performed to produce the linear ASL PDM.

Fig. 2 shows the two primary modes of deformation for the linear ASL PDM. These modes (eigenvectors) correspond to the largest eigenvalues of the training set and deform the model from the mean shape. By analysing the eigenvalues of the covariance matrix it can be determined that the first 30 eigenvectors (corresponding to the 30 largest eigenvalues) encompass 99.6% of the deformation within the model.

2.3. Applying non-linear constraints to shape space

To further constrain the model the approach presented in Refs. [2–4] is applied. Non-linear constraints to the model are added by performing cluster analysis on the dimensionally reduced data set after it has been projected down into PCA space. From the linear model it has been determined that the 30 primary modes encompass 99.6% of the deformation, by projecting each of the training vectors down into this lower dimensional space, a dimensional reduction of 400−30 is achieved. Cluster analysis is now performed upon the dimensionally reduced data set.

Fig. 3 shows the PCA space for the model as an orthographic projection into 3 dimensions for visualisation purposes, with the constraints shown as the bounding boxes (first two primary modes) of the linear patches (clusters) extracted via PCA. The bounded boxes are derived from the two most significant eigenvectors of the patch scaled to 3 standard deviations and represent the statistical bounds of that patch. The skew of the bounding boxes is due to the projection from 30 dimensions to 3. By constraining the model to lie within a linear patch the non-linearity of the shape space is estimated and a robust model produced.

Fig. 4 shows random shapes generated from within the allowable shape space of the linear ASL Model and the results of projecting these shapes into the allowable shape space of the CSSPDM. It can be clearly seen that the constrained model contains far less invalid deformation, and therefore, results in a more reliable model for tracking. Each random shape is also very close to a natural gesture in ASL and it is this correlation between cluster and gesture that can be used to perform gesture recognition.

3. A hybrid PDF for CONDENSATION

3.1. Least squares gradient descent tracking

From Fig. 3 it can be seen that shape space is segregated into at least two separate regions due to the movement of landmark points. Furthermore, connected patches of the model may not represent consistent movement of the model in the image frame. This leads to the model *jumping* between patches, even when within region 2. Under these circumstances it is not possible for the iterative refinement

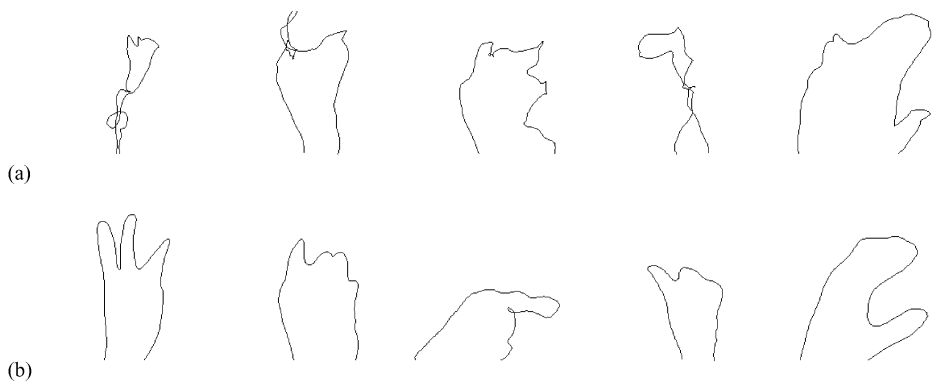

Fig. 4. Random shapes generated from within the ASL Models. (a) valid shapes generated from the Linear ASL PDM. (b) Valid shapes generated from the non-linear ASL CSSPDM.

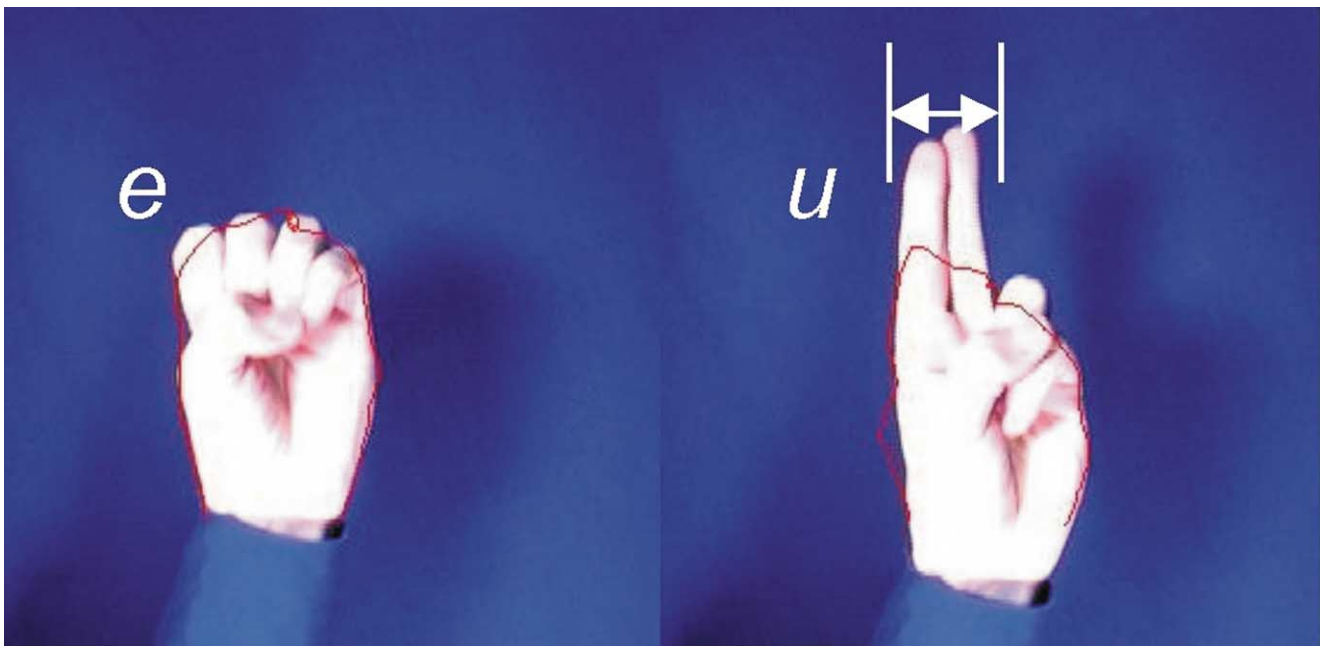

Fig. 5. ASM attempting to track an image sequence of the hand.

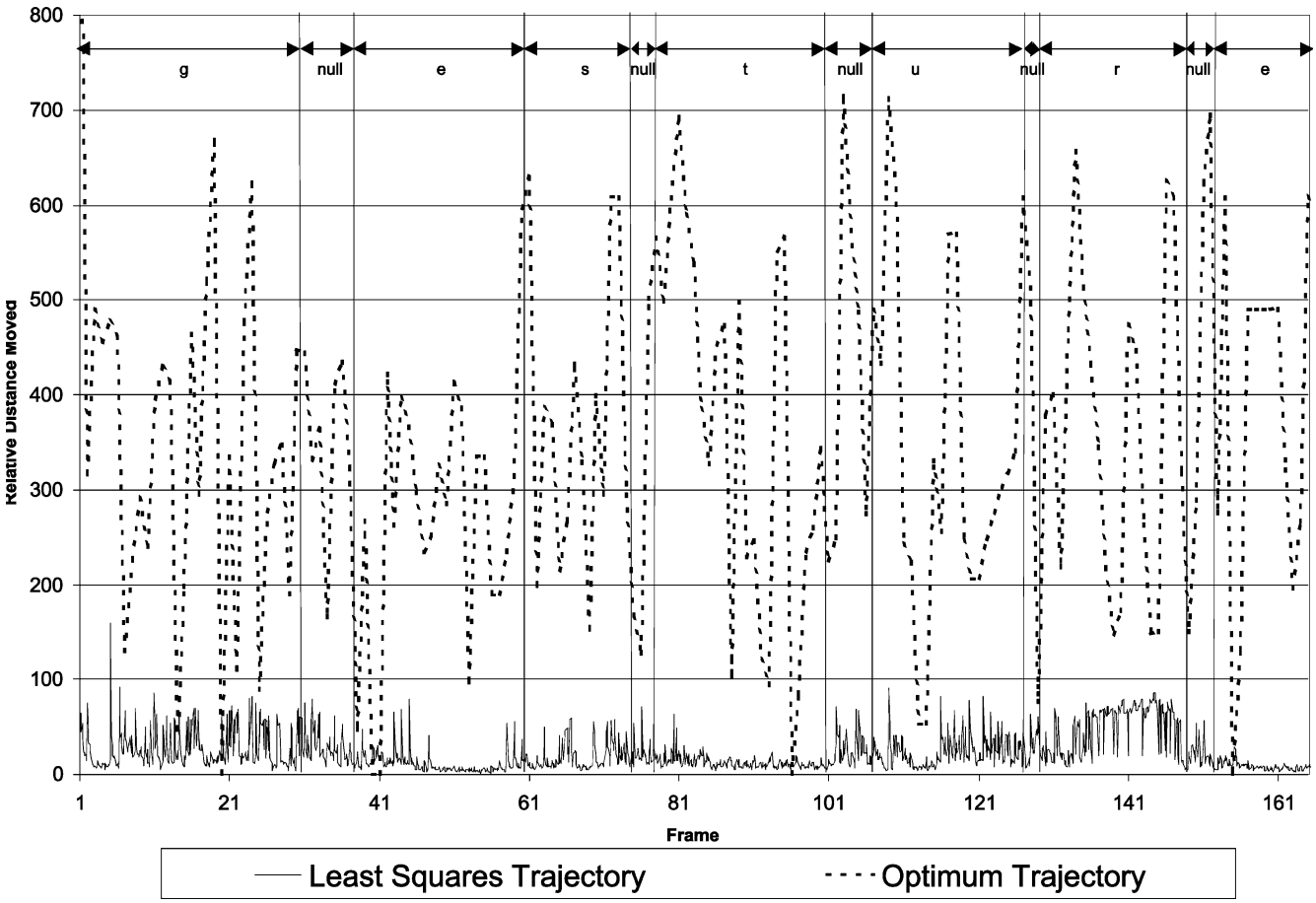

Fig. 6. Graph of distance moved at each iteration for least squares solution and optimum solution.

algorithm used for the classic PDM/ASM [5] to provide the '*jump*' between regions.

An image sequence was recorded of a hand signing the word 'gesture' which consisted of 170 frames. Throughout the remainder of this paper this sequence will be used for comparative evaluation. Fig. 5 shows the model attempting to track the image sequence through the transition from letters 'e' to 'u'. The model successfully tracks the letter 'e' but when the image sequence reaches the letter 'u' and the fingers elongate, the model is unable to make the jump to the new cluster responsible for modelling this letter. This problem is fundamental to the operation of the least squares iterative refinement algorithm and is due to two reasons:

1. Only a small section of the contour (marked in frame 'u') is responsible for 'pulling' the contour up to follow the elongated fingers. As this section is relatively small, compared to the remainder of the contour, it has less influence over the overall movement.
2. The maximum movement of the contour per iteration is governed by the length of the normal used to search around the contour. Hence this factor limits the distance the model can move through shape space at each iteration.

An obvious solution to these problems is to increase the search length along normals. However, larger normal searches allow the contour to affix to incorrect features in the image and hence results in degradation to tracking performance and additional computational complexity.

3.2. Finding the optimal ground truth for tracking

To locate the optimum solution (i.e. the closest allowable shape from the CSSPDM) for each iteration of the model, the space was exhaustively searched. If the assumption is made that any local patch of the CSSPDM can indeed be treated as a linear model, then the iterative refinement procedure can be used to move locally within that patch to the closest possible shape. Therefore, if the best match within each patch (cluster) is located for each frame, the resulting lowest cost solution must be the (near) optimum. This exhaustive search was performed on the 'gesture' image sequence. For every frame, each of the 150 clusters was assessed in turn. The mean shape of the cluster was used as a starting shape and the iterative refinement of the model, within the cluster, performed until the model converged (typically 40 iterations).

By analysing the optimum path through shape space and comparing this with the path taken by the least squares approach, the notion of discontinuity within shape shape can be confirmed.

Fig. 6 shows the distance moved through shape space at each iteration for both the optimum trajectory and the iterative refinement algorithm. The corresponding letters of the sequence are shown with the vertical lines denoting the approximate transition between letters. From this it can be clearly seen that the least squares iterative refinement algorithm makes small incremental movements at each iteration, whereas the optimum trajectory makes large 'jumps' at every frame. During the letters 'e' and 't' the least squares approach almost stops moving, which demonstrates that the model has converged upon a stable solution. However, the lack of such trends for other letters shows that the model is constantly struggling to better refine itself. Fig. 7 shows distance from the centre of shape space for the two trajectories. Again this demonstrates that the optimum path jumps violently within the space whereas the least squares approach makes small movements.

Note that in Fig. 7, the letter 'e' occurs twice during the sequence. However, during the first occurrence the least squares approach is at a distance of around 200 units from the mean whereas during the second occurrence it is at around 500. This demonstrates that there are at least two areas of shape space responsible for modelling the letter 'e' and these are distinctly separated in shape space. It also shows that the least squares approach can only use the local 'e' part of shape space and is incapable of jumping between them.

This confirms that not only is the non-linear shape space discontinuous but the least squares iterative refinement approach is incapable of providing a robust method for tracking. Instead a new method of applying the CSSPDM must be devised.

3.3. Supporting multiple hypotheses

Due to the discrete nature of the CSSPDM and the piecewise linear method of modelling non-linearity, the approach directly lends itself to a discrete PDF with the addition of a Markovian assumption. A first order model of temporal dynamics can be derived where the conditional probability $P(C_i^{t+1}|C_j^t)$ provides the probability that the model will move to cluster C_i given it was at C_j at the last time step. This conditional probability can be calculated from the training sequence and produces a 2D PDF of motion within shape space. The major discontinuities of the shape space occur when landmark points jump around the boundary and hence result in a jump in shape space (Figs. 6 and 7). However, within each patch, the model still makes small iterative movements. This can be confirmed by visualising the resulting PDF as a grey scale image.

Fig. 8 shows the ASL PDF, which has a heavy diagonal dominance. This dominance is when $\mathbf{argmax}_i(P(C_i^{t+1}|C_j^t))$ and $i = j$, i.e. the highest probability is that the PDM will stay within the present cluster. The assumption can, therefore, be made that within any local patch the model can iterate to a local solution. This confirms the assumption used when calculating the optimum model trajectory. This assumption also provides two benefits:

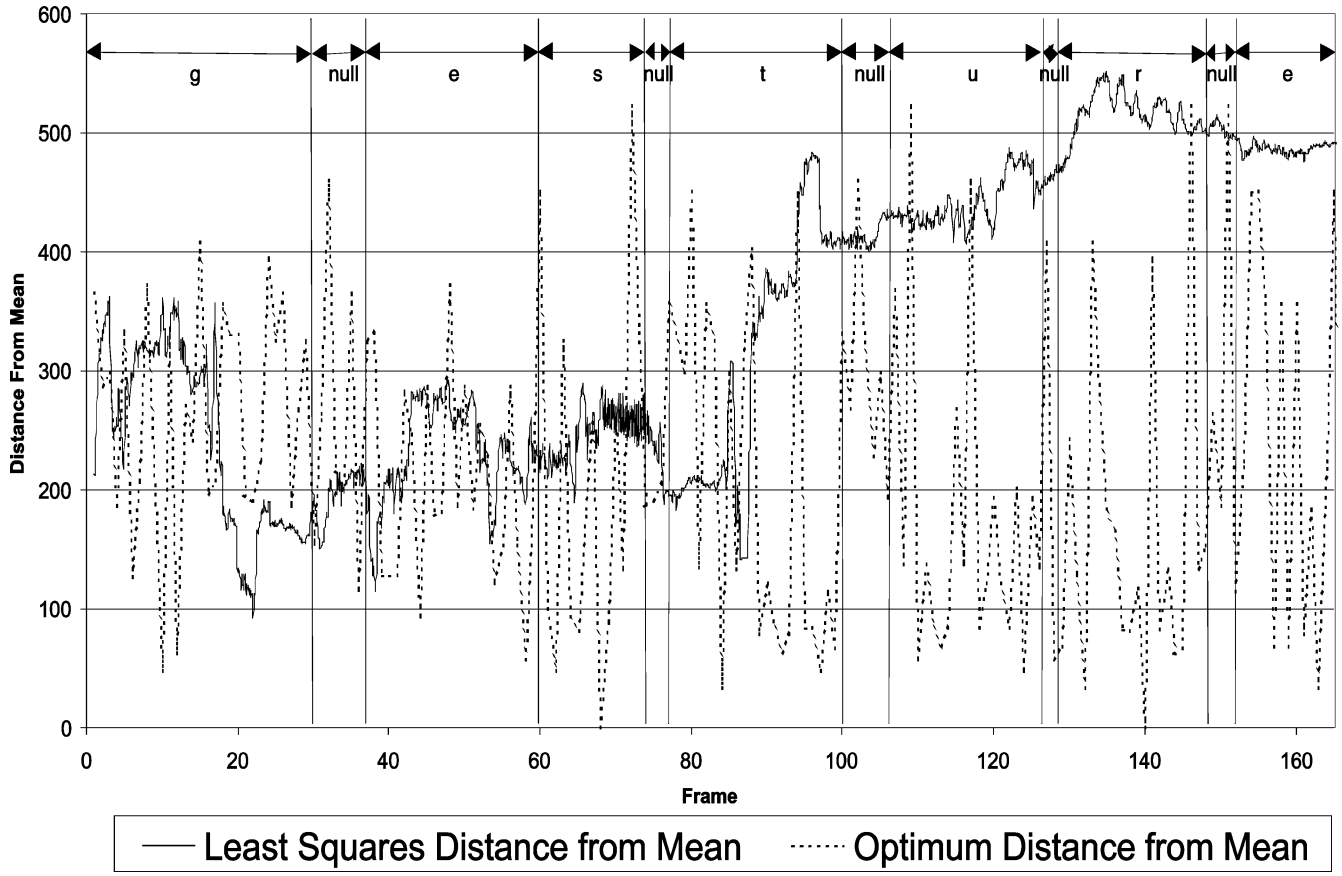

Fig. 7. Graph of distance from mean of shape space at each frame for least squares solution and optimum solution.

1. The iteration to convergence of any global optimisation technique can be enhanced by allowing each hypothesis to iterate to a better solution within the present cluster.
2. A smaller population is required, as only global differences in hypotheses need to be supported.

From the 'learnt' probability density function, a sample population can be generated at each iteration of the model. Given a good initialisation of the model and the associated cluster $C^{t=0}$, this can then be used to predict the future movement.

However, this approach, unlike condensation, does not recover well from failures. As the new population is solely based upon the current best-fit cluster, the approach is highly sensitive to both an accurate PDF and a good fit to the current object pose. To help overcome this drawback less emphasis must be placed upon the current best-fit hypothesis being the optimum (and hence correct) solution, thus providing more robustness to failure. This can be addressed by creating a new population of hypotheses, not from the current best fit model, but from the weighted sum of the best n hypotheses, described thus:

Algorithm 1. Weighted condensation

- From the PDF $P(C_i^t|C_j^{t-1})$, extract the probability vector $P(C_i^{t=1})$, which is the probability distribution of the first iteration, given $C_j^{t-1} = C^{t=0}$.
- Generate a randomly sampled distribution of k hypotheses $x_\rho[\rho = 1,\ldots,k]$, where $\mathbf{x}_\rho$ is the mean shape of cluster C_i and $P(C_i) = P(C_i^{t=1})$
  - While still tracking, i.e. while best hypothesis cost function is below a threshold t,
    - Fit k hypotheses, applying CSSPDM constraints and assess fitness using error metric
    - Sort hypotheses into descending order according to error
    - Iteratively refine first n hypotheses and resort
    - Apply the CSSPDM constraints and determine the n clusters C_η^{t-1}, where $\eta = 1,\ldots,n$ which produce the lowest error
    - From the PDF $P(C_i^t|C_j^{t-1})$, extract the vector $P(C_i^t)_\eta$ using the n extracted clusters. Take the weighted sum using a Gaussian weighting to form a new distribution $P'(C_i^t)$, where

$$P'(C_i^t) = \sum_{\eta=1}^{n} \omega_\eta P(C_i^t)_\eta,$$

$$\text{and } \omega_\eta = \exp\left[\frac{-9(1-\eta)^2}{2n^2}\right]$$

– Normalise the probability distribution $P'(C_i^t)$
– Generate a new random population of k hypotheses from the distribution $P'(C_i^t)$.

4. Extending temporal dynamics

It has been described how, with the addition of a first order Markov chain to the CSSPDM, a variation on the condensation algorithm can be used to provide robust tracking where either:

- The non-linearity of the PDM along with the discrete representation of the non-linear approximation leads to a discontinuous shape space.
- Rapid movement of the object within the image can produce large changes in the model parameters and hence large movements in shape space.

This Markovian model of dynamics can be used to explicitly constrain the movement of the model within shape space, or implicitly, using the variation on the CONDENSATION approach as previously shown. However, the use of temporal constraints relies upon the assumption that the training set from which the model is built, contains a thorough representation of all-possible deformation and movement. For simple models this is often true. However, for ASL it is not, and it is important to ask the question, '*What exactly is the temporal model representing?*'

The ASL PDF represents two aspects of motion:

1. The non-linear representation of shape space, how the individual clusters relate and how the model moves throughout the space to form letters.
2. It also contains information about the English language and how letters relate to form words and sentences.

As the PDF encodes both of these attributes it must be constructed from a training set which has a good representation of how the model deforms and be representative of the English language. This is, however, infeasible. If the ASL image sequence used previously is considered, it took 165 frames to record the seven letter word 'gesture'. Konheim reported a statistical study where the 1-state transition probabilities of the English Language were determined using 67,320 transitions between two successive letters [9]. As the 165 frames previously used produced an average of 20 frames per letter, this would constitute a training set in excess of 1.3 million frames not including transitional shapes between letters. As each frame produces a training shape this results in a training set which is of infeasible size.

The current ASL PDF (see Fig. 8) contains valuable information about how the model moves within shape space, but due to the deficiency in training it does not contain sufficient information to accurately model the transitions between the letters of the English language. Fortunately, it is relatively simple to gain a transition matrix for the English language by analysing large samples of electronic text and calculating the 1-state transitions. What is required is a method of combining this knowledge of English into the ASL PDF, producing a more generic and accurate model for tracking and classification.

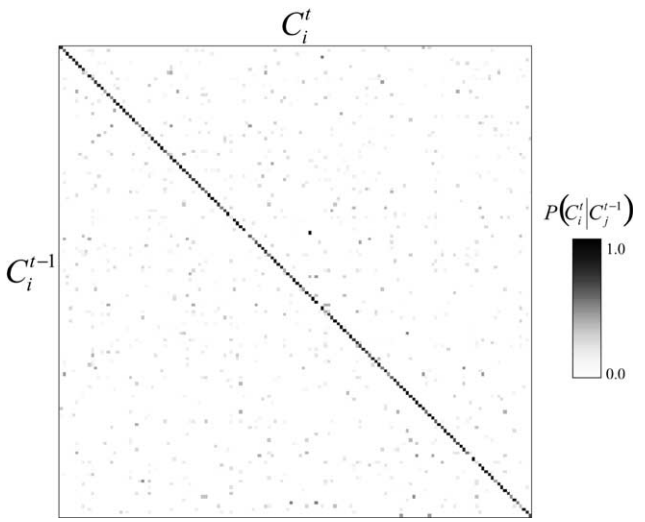

Fig. 8. Discrete probability density function for ASL Model.

4.1. The temporal model

The ASL PDF $P(C_i^t|C_j^{t-1})$, constructed from the training set, provides the probability that the model will move to cluster C_i given it was at cluster C_j at the last time step. Similarly a first order Markov Chain can be constructed for the English language which provides a new PDF $P(L_i^t|L_j^{t-1})$. Fig. 9 shows the PDF gained from this Markov Chain as taken from Konheim and shows the 1-state transitions calculated from a sample text of over 67 thousand letters [9].

Fig. 9 does not demonstrate a diagonal dominance, unlike Fig. 8. This is because the English language has few occurrences of repetitive letters in words whereas the previous PDF resulted from operations involving a high degree of repetition. The main trend that can be seen are the vertical stripes that occur for many of the letters. This shows letters, which have a high occurrence and are preceded by almost any other letter in the alphabet. The highest probabilities occur for the letter 'e' confirming that 'e' is the most commonly used letter in the English language. Another observation is the single transition from the row 'q' to the column 'u' as 'q' is always followed by a 'u' in standard English.

In order to incorporate this additional information learnt from sample text, a new ASL PDF must be constructed $P'(C_i^t|C_j^{t-1})$. To do this a mapping must be achieved which allows shape space to relate to gesture space.

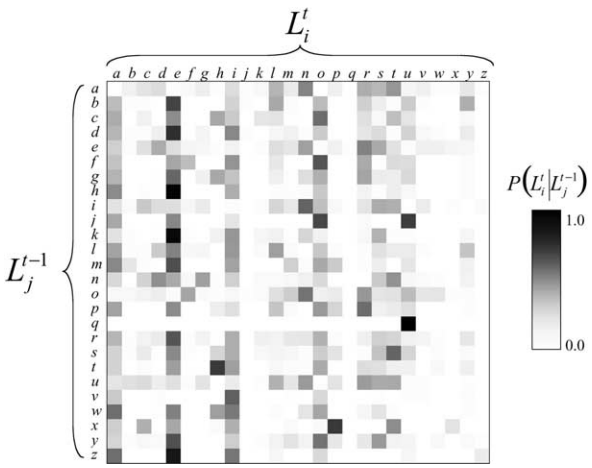

Fig. 9. Discrete probability density function for the English Language.

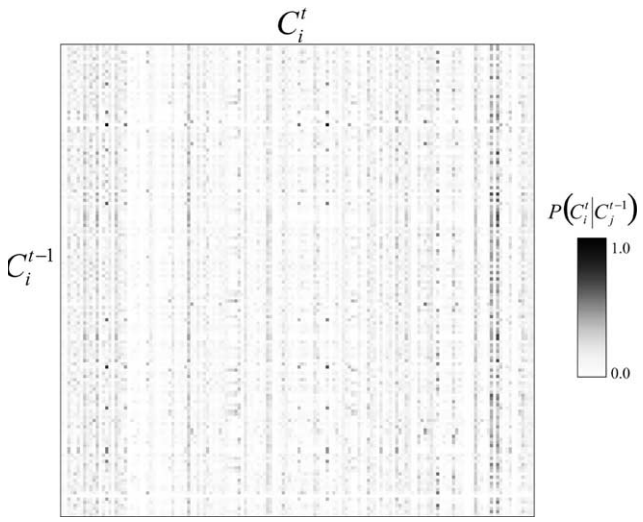

Fig. 10. Discrete probability density function for hybrid ASL Model.

4.2. Mapping between spaces

By labelling each training example with an associated letter a PDF can be generated which relates clusters in shape space to gestures. Here the conditional probability $P(L_i^t|C_j^t)$ provides a probability of the occurrence of a letter L given the model is in cluster C in shape space at any time. This conditional probability provides a mechanism to relate shape space to the gesture space where the constraints of the English language (as learnt) can be applied. However, for this to be of use, a method that allows this information to be mapped back into the shape space must be provided. This can be done using the common form of Bayes theorem where

$$P(C_i^t|L_j^t) = \frac{P(C_i^t)P\left(L_j^t\middle|C_i^t\right)}{P\left(L_j^t\right)}$$

However, where $P(C_i^t|L_j^t)$ and $P(C_i^t)$ can both be gained from the training set, $P(L_j^t)$ (the probability of the occurrence of a letter) can only be gained from analysing English text. As it is known that the training set does not fully represent the English Language this equation would lead to biasing of the final conditional probabilities. Instead, a variation of Bayes Theorem can be used, where

$$P(C_i^t|L_j^t) = \frac{P(C_i^t)P\left(L_j^t\middle|C_i^t\right)}{\sum P(C_i^t)P\left(L_j^t\middle|C_i^t\right)}$$

Using this form, $\sum P(C_i^t)P(L_j^t|C_i^t) \equiv P(L_j^t)$ but all probabilities are gained from the training set, and hence no bias occurs from mixing unrelated probabilities. This is possible as, although the training set does not contain a thorough representation of English, it does provide an accurate representation of the mapping between the two spaces.

4.3. The hybrid ASL PDF

A new ASL PDF can now be constructed which incorporates the 1-State transitions of the English language by treating the system like a Hidden Markov Model and projecting the transitions of the observation layer down into the Hidden (parameter space). Taking the current cluster of the model, the corresponding letter(s) associated with this cluster are determined and the 1-state transition matrix applied to extract the most likely next letter. The cluster(s) associated with this transition are then calculated,where

$$P'(C_i^t|C_j^{t-1}) = P(L_i^t|C_j^t)P(L_i^t|L_j^{t-1})P(C_i^t|L_j^t)$$

This produces a new ASL PDF which is shown in Fig. 10.

Fig. 10 demonstrates the same characteristic vertical strips seen from the English Language PDF, which it has inherited, and as such differs from the original ASL PDF in two ways:

1. Each cluster exhibits far more transitions to other clusters.
2. The diagonal dominance that is important to tracking, is missing.

Diagonal dominance can be forced upon the PDF by imposing diagonal dominance on either $P(L_i^t|L_j^{t-1})$ or $P'(C_i^t|C_j^{t-1})$. However, this is haphazard and risks over-biasing the hypothesis generated at each frame. An alternative is to simply ensure that the population generated at each step always includes at least one hypothesis from the current cluster.

Fig. 11 shows the results of three of the techniques discussed, namely that of the least squares gradient descent (ASM, algorithm [5]) the optimal solution gained through an exhaustive search of shape space and that of the hybrid condensation approach. The cost at each iteration is the sum of the pixel difference between the desired movement of the

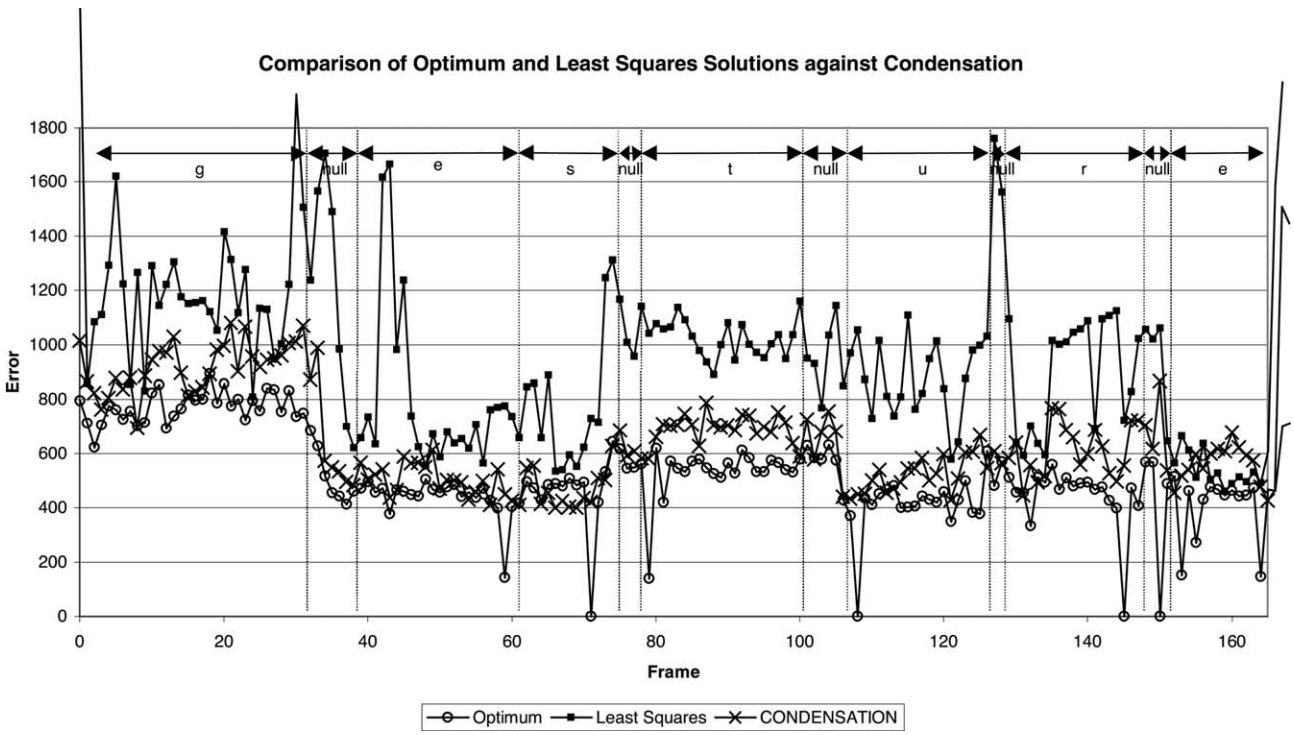

Fig. 11. Comparison of optimum and least squares solutions against hybrid CONDENSATION.

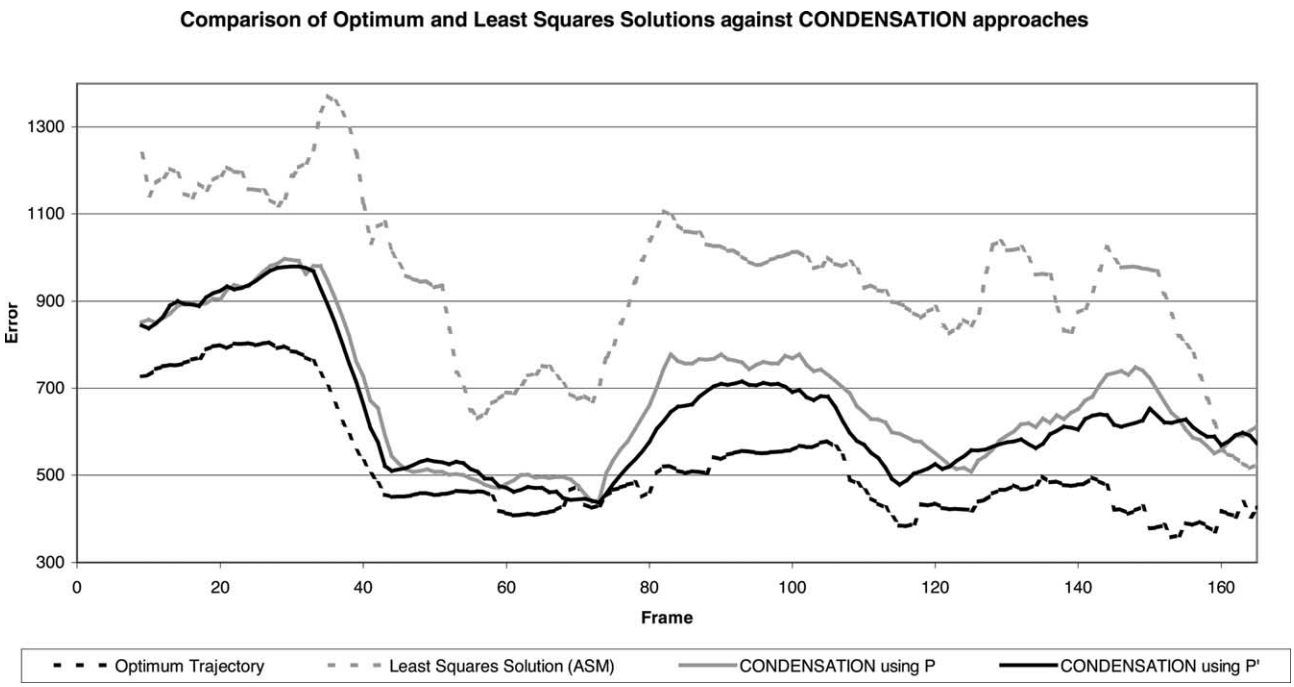

Fig. 12. Comparison of optimum and least squares solutions against hybrid CONDENSATION approaches using a rolling average.

model (gained from the assessment of the normals) and the final shape (after the constraints of the model have been applied) where low cost denotes a good model fit. Where multiple iterations per frame were performed, these are displayed as fractions of a frame to visualise the resulting error cost of iteration. It can clearly be seen that the optimum solution does indeed give the lowest results with the hybrid condensation producing only slightly higher error rates, both of which are significantly lower than those from the Least squares appraoch which fails catastophically.

Due to the clutered nature of the graph it is dificult to make any distinction as to the subtle differences between the use of $P(C_i^t|C_j^{t-1})$ (gained from the training set) and $P'(C_i^t|C_j^{t-1})$ (calculated in the pervious section). Using the same cost function as Fig. 11, Fig. 12 shows all four of the resulting error rates using a running mean of 10 samples to smooth out the plots and help simplify the graph so a visual comparison can be made. Here it can be seen that the CONDENSATION approach does indeed provide superior performance to that of the gradient descent method. Furthermore, it can clearly be seen that the adapted CONDENSATION approach based upon the newly formulated PDF that incorporates the English Language provides increased performance to that of the PDF generated from the training set alone.

5. Conclusions

This paper has presented the augmentation of statistical models with temporal dynamics gained through the probabilistic analysis of the training set and how this relates to movement within shape space. It has been shown how the discrete segregation of shape space used in the CSSPDM directly lends itself to a Markov chain approach to modelling temporal dynamics. It has been shown that the nature of shape space is often complex and discontinuous and how, using these additional learnt temporal constraints, tracking can be improved by supporting a population of multiple hypotheses. However, the key to this paper is the ability to project observation probabilities into a hidden shape space using an approach akin to a Hidden Markov Model where the simple acquisition of observation layer transitions can be propagated into the hidden parameter space to overcome the inadequacies of training. It has been shown how, using a hybrid CONDENSATION tracker, successful tracking can be achieved while maintaining a considerable lower population size to that of standard CONDENSATION. This approach has been aplied to a number of other image sequences and has demonstrated that it consistently produces better reults than either standard condensation using the pdf gained from the training set or the ASM algorithm.

References

[1] A. Blake, M. Isard, Active Contours, Springer, Berlin, 1998.
[2] R. Bowden, T.A. Mitchell, M. Sahardi, Cluster based non-linear principal component analysis, IEE Electronics Letters 33 (22) (1997) 1858–1858. 23rd October.
[3] R. Bowden, T.A. Mitchell, M. Sahardi, Non-linear Statistical Models for the 3D Reconstruction of Human Pose and Motion from Monocular Image Sequences, Image and Vision Computing 18 (9) (2002) 729–737.
[4] R. Bowden, Learning non-linear models of Deformation and Motion, PhD Thesis, Brunel University, Uxbridge, Middlesex, UB8 3PH, UK, 2000.
[5] T.F. Cootes, C.J. Taylor, D.H. Cooper, J. Graham, Active shape models—their training and application, Computer Vision and Image Understanding 61 (1) (1995) 38–59.
[6] A. Hill, C.J. Taylor, Model based image interpretation using genetic algorithms, Proceedings British Machine Vision Conference, Springer, Berlin, 1991, pp. 266–274.
[7] A. Hill, C.J. Taylor, Model based image interpretation using genetic algorithms, Image Vision Computing. 10 (1992) 295–300.
[8] M. Isard, A. Blake, Condensation—conditional density propagation for visual tracking, International Journal of Computer Vision (1998).
[9] A.G. Konheim, Cryptography: a primer, Wiley, New York, 1982.
[10] C. Vogler, D. Metaxas, ASL recognition based on a coupling between HMMs and 3D motion analysis, Proceedings of the International Conference on Computer Vision, Mumbai, India (1998) 363–369. 4–7 January.
[11] T. Starner, A. Pentland, Visual recognition of american sign language using hidden Markov models, International Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland (1995) 189–194.
[12] W. Gao, Enhanced user by hand gesture recognition, CHI-95 Workshop on User Interface by Hand Gesture, Denver (1995) 45–53.
[13] C. Uras, A. Verri, Sign language recognition: an application of the theory of size functions, Sixth British Machine Vision Conference 2 (1995) 711–720.
[14] M. Handouyahia, D. Ziou, S. Wang, Sign language recognition using moment-based size functions, Vision Interface '99, Trois-Rivieres, Canada (1999) 19–21 May.
[15] R. Watson, A survey of gesture recognition techniques, Technical Report TCD-CS-93-11, Department of Computer Science, Trinity College, Dublin 2. July 1993.
[16] J.A. Bangham, S.J. Cox, M. Lincoln, I. Marshall, M. Tutt, M. Wells, Signing for the deaf using virtual humans, IEE Seminar on Speech and language processing for disabled and elderly people London, April, 2000. www.visicast.sys.uea.ac.uk/.
[17] W. Freeman, M. Roth, Orientation histograms for hand gesture recognition, International Workshop on Automatic Face and Gesture Recognition, Switzerland (1995).