Joint Manifold Distance: a new approach to appearance based clustering

Andrew W. Fitzgibbon and Andrew Zisserman
Robotics Research Group, Department of Engineering Science,
University of Oxford, United Kingdom
http://www.robots.ox.ac.uk/˜vgg

Abstract

We wish to match sets of images to sets of images where both sets are undergoing various distortions such as viewpoint and lighting changes.

To this end we have developed a Joint Manifold Distance (JMD) which measures the distance between two subspaces, where each subspace is invariant to a desired group of transformations, for example affine warping of the image plane. The JMD may be seen as generalizing invariant distance metrics such as tangent distance in two important ways. First, formally representing priors on the image distribution avoids certain difficulties which, in previous work, have required ad-hoc correction. The second contribution is the observation that previous distances have been computed using what amounted to "home-grown" nonlinear optimizers, and that more reliable results can be obtained by using generic optimizers which have been developed in the numerical analysis community, and which automatically set the parameters which home-grown methods must set by art.

The JMD is used in this work to cluster faces in video. Sets of faces detected in contiguous frames define the subspaces, and distance between the subspaces is computed using JMD. In this way the principal cast of a movie can be 'discovered' as the principal clusters. We demonstrate the method on a feature-length movie.

1. Introduction

We would like to cluster instances of objects in a video in an unsupervised manner in order to 'discover' the significant characters, scenes, events etc. This requires that our measure of distance between two imaged instances is ideally invariant to the changes in viewpoint and lighting that affect the image—so that our clustering is of the object, not its image.

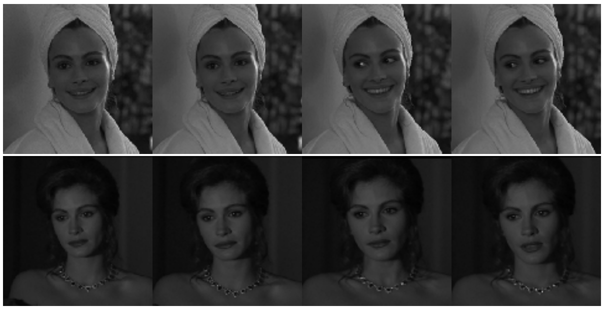

Figure 1: Matching image subspaces. Each row is a sequence of images spanning a subspace, and the goal is to determine for a pair of sequences, whether the subspaces spanned by the sequences are the same. The images within each subspace are registered, but the transformation between the subspaces is unknown. The distance between subspaces must be invariant to the unknown registration between the sequences.

As an example of such clustering, in this paper our objective is to establish matches between the faces that occur throughout a feature length movie. This is a very challenging problem: a film typically has 100-150K frames; and in addition to changes of lighting and viewpoint, faces also change expression and are partially occluded—for example by hands, telephones or spectacles. In movies in particular, lighting and viewpoint are intentionally dramatically varied. This makes the clustering problem significantly more difficult than in traditional "mugshot" applications.

One way to proceed is to construct a distance function

$d(x_1, x_2)$ between two images instances x_1 and x_2 which is invariant to all such perturbations and deformations that occur. For example invariance to viewpoint can be achieved (to a first approximation) by designing the distance function to be invariant to an affine transformation of x_i, so that for example $d(x_1, x_2) = d(x_1, T(x_2; \mathbf{a}_2))$, where T represents the affine transformation of the image. Implementation of these measures via tangent distance and its extensions [3, 5, 15, 16] allows efficient computation for classes of parametrized transformations. This idea can be extended to any desired transformation, e.g. for photometric changes or changes in expression, provided a parametrized model of the class of transformations is available.

An alternative is to define a distance function $d(x, S)$ between a point x and a (possibly infinite) set of points S, where S contains exemplars of the perturbations and deformations. Often this reduces to the distance between a point and a linear subspace. For example consider photometric invariance. A widely used approximation is that (under restricted conditions of no shadowing, Lambertian reflectance etc), the space of all images under all lighting is spanned by a four dimensional linear space [1, 13]. Higher dimensional spaces can approximate other illumination effects such as self-shadowing (attached shadows) [2]. So photometric invariance could be achieved if S is the space of lighting images, e.g. acquired by an SVD of many registered images [17]. Given training examples which exercise these variations, a transformation-aware principal component analysis [7, 14] can compute the subspace even in the case of unregistered sets of images.

A third approach is to remove the variations before matching, by projecting each image onto a space which does not permit such deviations. For example, the photometric variation problem can often be avoided by filtering the images (e.g. by a high pass filter) to significantly ameliorate lighting effects, effectively collapsing the space to a much lower dimension. In the limit the space is collapsed to a single point and the approach reduces to computing a point to point distance $d(x_1, x_2)$.

Here we partition the deformations into those that can be modelled or removed to some approximation (viewpoint by affine transformations, photometric filtering), and use a set of images to span the residual: the parts of viewpoint not modelled by affinity, errors in computing registration, and—in the face case—changes in expression. Figure 2 shows a set before and after affine registration. In video sets of this type are readily available because objects do not arbitrarily disappear between contiguous frames, but can be easily tracked so that clustering over consecutive frames is straightforward [8].

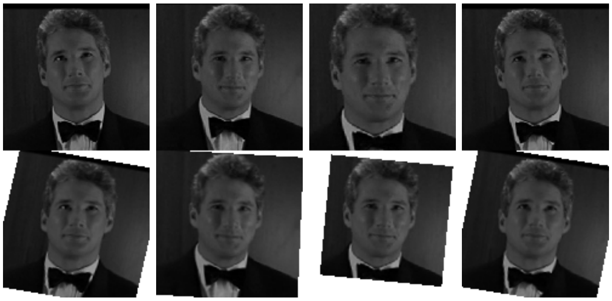

Figure 2: Affine registration. (Top) Sequence of faces obtained by the face detector. (Bottom) Affine registered sequence. There is an overall unresolved affine transformation which must be accounted for when comparing this with other such sequences.

There is one further development that is clearly motivated by figure 1: to determine the distance between two finite sets of size n, it is not necessary to compute the n^2 distances between each point in one set and each in the other—instead the distance between the sets $d(S_1, S_2)$ can be measured directly. This paper explores these three distance functions, $d(x_1, x_2), d(x, S), d(S_1, S_2)$ including an extension of incorporating learnt priors on the various transformations, describes efficient implementations, and demonstrates their performance in the face clustering application.

2. Classes of distance functions

In this section we discuss the three classes of distance function: point to point, point to set, and set to set. First, however, let us consider the observation model.

We have samples x each associated with a "true" datum $\tilde{x}$ which are drawn independently from the density described by $p(x|\tilde{x})$. Given observations x_1 and x_2 our objective is to determine the likelihood $p(x_1, x_2)$ that both

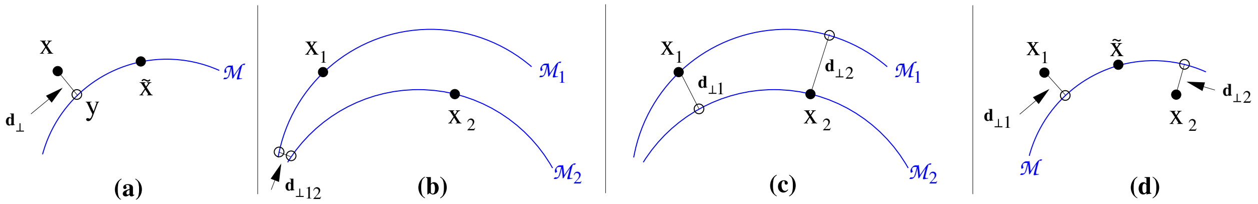

Figure 3: Several definitions of manifold distance between sample points x_1 and x_2. The distance from a datum to the hidden 'true' point $\tilde{x}$ is measured as the distance to the manifold $\mathcal{M} = \{T(\tilde{x}; \mathbf{a}) \forall \mathbf{a} \in \mathbb{R}^m\}$. (a) **Transfer distance**. When $\tilde{x}$ is approximated by one of the sample points, here x_2, this is the "one-sided" manifold distance. (b) **Two-sided manifold distance**. This definition can sometimes make distances between disparate objects arbitrarily small, for example by mapping each image to a single point. (c) **Symmetric transfer distance**. Sometimes called "two-sided" manifold distance. (d) **Manifold distance**. The hidden variable $\tilde{x}$ is explicitly included, so the manifold to which distance is measured must move during the optimization. Section 2.1 shows how to compute a tangent approximation which accounts for the manifold movement.

are samples of the same $\tilde{x}$:

$$p(x_1, x_2) = \int d\tilde{x}\, p(x_1|\tilde{x}) p(x_2|\tilde{x}) p(\tilde{x}) \qquad (1)$$

where $p(\tilde{x})$ is the prior distribution on $\tilde{x}$.

An observation x is generated by applying a transformation $\mathbf{a}$ to a true datum $\tilde{x}$ and then adding noise. The family of transformations is parametrized by a vector of parameters $\mathbf{a}$, and the transformation is given by

$$x \rightarrow T(x; \mathbf{a}).$$

For example, if the observations x are n-D points then for a transformation for scaling about the origin, $\mathbf{a}$ will have exactly one element a_1 representing the scale and points will transform as $T(x; \mathbf{a}) = x a_1$. The density $p(x|\tilde{x})$ may be expanded as

$$p(x|\tilde{x}) = \int d\mathbf{a}\, p(x|\mathbf{a}, \tilde{x}) p(\mathbf{a}|\tilde{x}) \qquad (2)$$

where $p(\mathbf{a}|\tilde{x})$ is the prior probability of the transformation $\mathbf{a}$ given $\tilde{x}$, and it will be assumed that $p(\mathbf{a}|\tilde{x}) = p(\mathbf{a})$, i.e. that the prior is independent of the datum $\tilde{x}$. The *manifold* itself is encoded in the prior $p(\mathbf{a})$. If this prior were completely unrestricting, we would have the standard picture of the manifold as a subset of the space of images in which x lives, with the dimensionality of the manifold equal to that of the parameters $\mathbf{a}$. For an affine transformation of n-pixel images, this would be a six-dimensional manifold in $\mathbb{R}^n$. In practice, it is important to place priors on $\mathbf{a}$, so some points on that manifold are less likely to be observed than others. For example, we might expect that the transformation which shrinks the image down to less than one pixel square is unlikely. Using (2), the joint likelihood (1) may thus be expanded as

$$\begin{aligned} p(x_1, x_2) = \int d\tilde{x}\, d\mathbf{a}_1\, d\mathbf{a}_2 \ldots \\ \ldots p(x_1|\tilde{x}, \mathbf{a}_1) p(x_2|\tilde{x}, \mathbf{a}_2) p(\mathbf{a}_1) p(\mathbf{a}_2) p(\tilde{x}) \end{aligned}$$

The terms in this expression are summarized as follows: $p(x_1|\tilde{x}, \mathbf{a}_1)$ is the likelihood of x_1, given the true point $\tilde{x}$, transformed by the transformation parameters $\mathbf{a}_1$. The likelihood of x_2 is $p(x_2|\tilde{x}, \mathbf{a}_2)$. $p(\mathbf{a}_i)$ is the prior probability of transformation $\mathbf{a}_i$—this will be estimated from training examples. $p(\tilde{x})$ is the prior distribution on the true point $\tilde{x}$. Here we set this to a broad Gaussian, which yields a term analogous to the "spring" tangent distance regularizer [15].

It will be assumed here that the image likelihoods can be approximated by a distribution whose density function is of the form $p(x|\tilde{x}, \mathbf{a}) = e^{-\rho(z)}$ where z is the difference image $x - T(\tilde{x}; \mathbf{a})$, and ρ is a kernel function. Choices of the kernel ρ include the Gaussian model $\rho(z) = \|\Sigma^{-\frac{1}{2}} z\|^2$ or a robust distribution, and the choices will be discussed later in the paper.

The MAP estimate is obtained from the joint likelihood as:

$$\begin{aligned} p(x_1, x_2) &\approx p_{MAP}(x_1, x_2) \\ &= \max_{\mathbf{a}_1,\mathbf{a}_2,\tilde{x}} p(x_1|\tilde{x}, \mathbf{a}_1)p(x_2|\tilde{x}, \mathbf{a}_2)p(\mathbf{a}_1)p(\mathbf{a}_2)p(\tilde{x}) \end{aligned}$$

and then the distance is defined as the negative log likelihood $d(x_1, x_2) := -\log p_{MAP}(x_1, x_2)$. We will refer to this as the *manifold distance* between two points.

Having derived manifold distance from a generative model as above, we relate it to the several different definitions in the literature. The primary distinction made is between "one-sided" and "two-sided" distances, but we show here that neither is equivalent to the true manifold distance. The discussion is clearer if the manifold distance is rewritten as a sum of negative log likelihoods [1]

$$\begin{aligned} d(x_1, x_2) = \min_{\mathbf{a}_1,\mathbf{a}_2,\tilde{x}} \quad & E(x_1 - T(\tilde{x}; \mathbf{a}_1))+ \\ & E(x_2 - T(\tilde{x}; \mathbf{a}_2))+ \\ & E(\mathbf{a}_1) + E(\mathbf{a}_2) + E(\tilde{x}). \end{aligned}$$

Computation of the true manifold distance includes an optimization over the hidden variable $\tilde{x}$ as well as the transformation parameters $(\mathbf{a}_1, \mathbf{a}_2)$. In the case of image matching, $\tilde{x}$ is the underlying true image which is warped and noise-corrupted to give the captured images x_1 and x_2. A number of alternative definitions in the literature have eliminated $\tilde{x}$ in various ways.

Variations on manifold distance: In the first, illustrated in figure 3a, $\tilde{x}$ is chosen to be equal to one of the two data points, say x_1. The manifold distance of the second point x_2 is then

$$d_1(x_1, x_2) = \min_{\mathbf{a}} E(x_2 - T(x_1; \mathbf{a}))$$

This formulation—called "transfer" or "one-sided" distance—is easy to compute, but has the disadvantage that it causes $d(.,.)$ to fail to be a metric, as $d_1(x_1, x_2) \neq d_1(x_2, x_1)$. It also means that priors on $\tilde{x}$ cannot be incorporated, as $\tilde{x}$ is fixed to be one of the data points.

Symmetry is addressed by defining the "two-sided" distance (figure 3b)

$$d_2(x_1, x_2) = \min_{\mathbf{a}_1,\mathbf{a}_2} E(T(x_2; \mathbf{a}_2) - T(x_1; \mathbf{a}_1)) \qquad (3)$$

in which both images are transformed before comparison. However this can yield spurious solutions, the canonical example being that images under affine transformations can be mapped to a single point by scaling, yielding a low distance for any pair (x_1, x_2). A variant that does not suffer this collapse, but appears not to be widely used, is the "symmetric transfer distance" (figure 3c)

$$d_s(x_1, x_2) = \min_{\mathbf{a}_1,\mathbf{a}_2} E(x_2 - T(x_1; \mathbf{a}_1)) + E(x_1 - T(x_2; \mathbf{a}_2)).$$

Again, however, priors on $\tilde{x}$ are not readily included. We show in this paper that these approximations are not necessary, and that the general form $d(.,.)$ may be optimized over $\tilde{x}$ and the transformations $(\mathbf{a}_1, \mathbf{a}_2)$ directly.

The effect of priors: Suppose for the moment there are no priors on the transformation, and that $\tilde{x}$ is known. Then in the single-sided case, the distance is minimized by the closest point to x on the orbit through $\tilde{x}$. Similarly the manifold distance is minimized by the points closest to x_1 and x_2 respectively on the orbit through $\tilde{x}$, but there is a freedom (symmetry) to choose $\tilde{x}$ as any point on the orbit, since it is only the orbit that defines the distances. Introducing the priors on the transformation breaks this symmetry—the point $\tilde{x}$ must now lie "between" x_1 and x_2 in order to reduce the prior terms $E(\mathbf{a}_1)+E(\mathbf{a}_2)$. Also the estimated point x is no longer given by the closest point on the orbit.

Discussion: Many existing variants amount to supplying different forms for the priors and likelihoods, although to our knowledge, no work has included all simultaneously, or optimized over $\tilde{x}$. Previous authors have added priors to the one and two sided distances. The regularizing term in Schwenk *et al*'s constraint tangent distance[11] may be seen as imposing a uniform prior using this term, and Keysers *et al*'s probabilistic tangent distance [6] shows how a Gaussian prior can be included. Jojic et al [5] derive the form of the two-sided tangent distance (3) with priors on the transformation parameters, but do not include priors on $\tilde{x}$. They do include priors on $\tilde{x}$ in a generative model learning framework, not in the distance function, but even there only draw $\tilde{x}$ from a discrete set of cluster centres and assume zero variance.

2.1. Computing the point distance

Approximating $T(x; \mathbf{a})$ by a first-order Taylor expansion converts the manifold distance into the tangent distance.

[1] To avoid clutter, there is an overloading of notation here: $E(\cdot)$ is not the same negative log-likelihood function for each variable, but indicates that the appropriate likelihood for the argument type is being computed as above.

Specifically, if $\mathbf{a}$ is the m-dimensional parameter vector, then the transformation of point x under transformation $\mathbf{a}$ is

$$\begin{aligned} T(x;\mathbf{a}) &\approx x + \frac{\partial T}{\partial a_1}(x;\mathbf{a})a_1 + \cdots + \frac{\partial T}{\partial a_m}(x;\mathbf{a})a_m \\ &= x + \mathbf{La} \end{aligned}$$

where the columns of L are the derivatives of the transformation at x. If x itself is unknown, L is often approximated by computing tangents at a convenient nearby point (of which more in §3).

In the case of Gaussian priors, a solution to the minimization in $d(x_1, x_2)$ can be obtained directly. The equation to be minimized is

$$\begin{aligned} &\min_{x,a_1,a_2} \\ &\quad \underbrace{|x_1 - (x + L_1 a_1)|^2}_{-\log p(x_1|x,a_1)} + \underbrace{|x_2 - (x + L_2 a_2)|^2}_{-\log p(x_2|x,a_2)} + \\ &\qquad\qquad \underbrace{|D[a_1 a_2]^\top + d|^2}_{-\log p(a_1)p(a_2)} + \underbrace{|Sx + s|^2}_{-\log p(x)} \end{aligned}$$

where D and d encode the parameters of a single normal distribution describing the prior probability of the transformation parameters, and S and s represent the prior on the unwarped image x. Specifically, if the prior on the transformation parameters is $\mathcal{N}(\Sigma_a, \mu_a)$, then $D = \Sigma_a^{-\frac{1}{2}}$ and $d = -D\mu$. For clarity, the pixel values x are assumed to have been scaled so that their noise is drawn from a unit-variance Gaussian per pixel, although spatially varying noise is easily incorporated.

Gathering the terms to be minimized into a single vector $\mathbf{x} = [x, a_1, a_2]^\top$ gives the quadratic form

$$\begin{aligned} &\min_{x,a_1,a_2} \left| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} I & L_1 & 0 \\ I & 0 & L_2 \end{pmatrix} \begin{pmatrix} x \\ a_1 \\ a_2 \end{pmatrix} \right|^2 + \\ &\qquad \left| \begin{pmatrix} S & 0 & 0 \\ 0 & D_1 & 0 \\ 0 & 0 & D_2 \end{pmatrix} \begin{pmatrix} x \\ a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} s \\ d_1 \\ d_2 \end{pmatrix} \right|^2 \\ &= \min_{x,a_1,a_2} \left| \begin{pmatrix} I & L_1 & 0 \\ I & 0 & L_2 \\ S & 0 & 0 \\ 0 & D_1 & 0 \\ 0 & 0 & D_2 \end{pmatrix} \begin{pmatrix} x \\ a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} -x_1 \\ -x_2 \\ s \\ d_1 \\ d_2 \end{pmatrix} \right|^2 \end{aligned}$$

This is of the form $\min_{\mathbf{z}} |Gz + g|^2$ for which a closed-form solution is readily found. Naively implemented, this would be computationally very expensive, requiring the pseudo-inversion of a matrix whose side length is of the order of the number of pixels. However, the special structure of G means that the minimum can be computed with no more complexity than the two-sided tangent distance.

2.2. Point to subspace distance

A linear subspace of images is defined by a mean image $\mathbf{m}$ and a set of basis vectors $\mathtt{M}$. An image in the space is linearly parametrized by vector $\mathbf{u}$ yielding the set

$$S = \{\mathbf{m} + \mathtt{M}\mathbf{u} \mid \mathbf{u} \in \mathcal{U}\}$$

The distance from a query point x_1 to the space is then

$$\begin{aligned} d(x_1, S) &= \min_{y \in S} d(x_1, y) \\ &= \min_{\mathbf{u}} \|\mathbf{m} + \mathtt{M}\mathbf{u} - x_1\|^2 \end{aligned}$$

which is easily computed as minimization of a quadratic form. In real examples, y will be subject to an unknown transformation $T(y, \mathbf{a})$, and there will be priors on $\mathbf{a}$ and $\mathbf{u}$. Adding these terms gives the one-sided point-to-subspace distance

$$d(x_1, S) = \min_{\mathbf{u},\mathbf{a}} \|T(\mathbf{m} + \mathtt{M}\mathbf{u}; \mathbf{a}) - x_1\|^2 + E(\mathbf{a}) + E(\mathbf{u})$$

Note here that the prior on $\mathbf{u}$ acts as a prior on the latent image y. Denoting this prior by $p(y)$, the subspace distance becomes

$$d(x_1, S) = \min_{y,\mathbf{a}} \|T(y; \mathbf{a}) - x_1\|^2 + E(\mathbf{a}) - \log p(y)$$

which is an analogue of the one-sided manifold distance where x_1 is drawn from the prior distribution over y.

2.3. Distance between subspaces

Finally, the problem which faces this paper is to compute the distance between two subspaces. Defining subspaces S and T as

$$\begin{aligned} S &= \{\mathbf{m} + \mathtt{M}\mathbf{u} \mid \mathbf{u} \in \mathcal{U}\} \\ T &= \{\mathbf{n} + \mathtt{N}\mathbf{v} \mid \mathbf{v} \in \mathcal{V}\}, \end{aligned}$$

then probability density from which examples are drawn is defined by priors over the parameter vectors $\mathbf{u}$ and $\mathbf{v}$. These distributions in turn induce distributions of images in the image space, denoted $p(s)$ and $p(t)$, say. In previous work, Shakhnarovich *et al* [12] define the distance between these subspaces as the Kullback-Leibler divergence between the image-space distributions, but do not incorporate the transformation manifold. Incorporating this manifold is the subject of the next section.

3. Joint manifold distance

Finally, the problem which faces this paper is to compute the distance between two subspaces, represented as above, where points in the subspaces may be subject to an unknown image-space transformation parametrized by parameter vectors $\mathbf{a}$. Defining subspaces S and T as

$$\begin{aligned} S &= \{\mathbf{m} + \mathtt{M}\mathbf{u} \mid \mathbf{u} \in \mathcal{U}\} \\ T &= \{\mathbf{n} + \mathtt{N}\mathbf{v} \mid \mathbf{v} \in \mathcal{V}\}, \end{aligned}$$

the joint manifold distance is defined as the infimum of manifold distance between points in the two subspaces.

$$d(S,T) = \min_{x \in S, y \in T} d(x,y).$$

Including priors on the transformation parameters $\mathbf{a}$ and $\mathbf{b}$, and on the parameter vectors $\mathbf{u}$ and $\mathbf{v}$, and expressing in terms of negative log likelihoods yields

$$\begin{aligned} &d(S,T) = \\ &\quad \min_{\mathbf{u},\mathbf{v},\mathbf{a},\mathbf{b}} \|T(\mathbf{m} + \mathtt{M}\mathbf{u}, \mathbf{a}) - T(\mathbf{n} + \mathtt{N}\mathbf{v}; \mathbf{b})\|^2 + \\ &\qquad\qquad E(\mathbf{a}) + E(\mathbf{b}) + E(\mathbf{u}) + E(\mathbf{v}) \end{aligned} \quad (4)$$

This distance could be computed as above by computing the Taylor expansion of $T(\cdot,\cdot)$, yielding the subspace analogue of tangent distance. This superficially appears attractive, as it provides a closed-form solution to the minimization. However, as other authors have noted [3, 14, 16], the accuracy of the tangent distance depends on the point about which the Taylor expansion is performed, and practical implementations require iterative refinement of the computation. Incorporating this iterative refinement is equivalent to a Newton minimization of the original expression (4).

However, Newton optimization is but one of a panoply of available optimization techniques, and has been superseded in recent decades by several more effective techniques, notably the trust-region strategies derived from the original Levenberg-Marquardt algorithm [9]. In this work we leverage such strategies and directly minimize the various cost functions with no more reference to the linearizations than is implied by efficiently computing the derivatives. If the kernel ρ is quadratic, then the quadratic optimization problems which previous authors have solved explicitly (and which this paper solves in §2.1) are constructed implicitly within the nonlinear minimizer, which employs well honed bookkeeping strategies to ensure fast and accurate optimization. If ρ is a robust kernel, the linearizations are more complex than for the quadratic case, but this complexity is all encapsulated in the computation of the error Jacobian.

4. Practice

The application of the above principles to a hard real-world problem offers some useful insights into this class of techniques. The implementation issues which we have found to be most important are: the use of a high-quality nonlinear optimizer to estimate the distances, good choice of priors on transformation parameters, and careful preprocessing to mitigate the worst lighting effects. The application we consider is the automatic clustering of faces in feature-length movies. By running a face detector over the movie, we reduce the input to a set of several thousand faces. Temporal segmentation of the face sequences produces a few hundred sequences of 10 to 50 frames. Agglomerative clustering using the subspace to subspace distance reduces this to a small number of clusters, roughly corresponding to the cast list. The following sections discuss this application, dwelling on the areas of difficulty.

Face detection and processing: Faces were detected independently in every fifth frame of the input movie using a local implementation of the Schneiderman and Kanade detector [10]. The use of every fifth frame is purely in order to reduce the volume of data, detection in every frame would be preferable. Faces were sequentially matched using the manifold distance (see §2.1). A conservative global threshold on the distances yields a segmentation into sequences of the same character. In the test movie, 5582 faces were detected in 166510 frames, of which 2710 were in the 200 longest sequences. The transformation parameters recovered in the computation of manifold distance are used to align the faces to the first frame in the sequence.

In order to mitigate the effects of lighting variation, the faces are high-pass filtered by subtracting a Gaussian smoothed copy ($\sigma = 5$ pixels).

Priors and covariances: Prior probabilities for the transformation parameters were computed by manually identifying eyes and mouth centre in 200 detected faces. These were transformed to canonical points in the image and the variation of the six parameter affine transform modelled by a single Gaussian. This prior proved adequate for experiments on a wide range of sequences.

Error metric—robustness: A significant advantage of the explicit minimization is that noise distributions other than Gaussian can be assumed for the imaging process. In particular, a heavy-tailed distribution leading to a robust error kernel can be incorporated without difficulty. Here we use the Lorentzian $\rho(z) = \log(1 + \beta\|z\|^2)$. This confers significantly improved resistance to occlusion, most commonly caused in movies by objects such as telephones and hands in front of the face. The estimation is concentrated on the centre of the image by assigning a per-pixel variance which increases towards the edge of the image, thereby tending to ignore the image periphery.

Subspace construction: Given a set of tracked images, we wish to compute a subspace which spans these images, and allows some degree of extrapolation of the observed deformations. In our implementation, the image sets have been aligned using the manifold distance transform parameters, so principal components analysis will suffice to encapsulate the variation. To permit extrapolation, we augment the set of images with x and y spatial derivatives of the images [14] to allow some additional small deformations. Examples of mapping faces onto this subspace are shown in figure 4.

Clustering: Clustering using the sequence-to-sequence distance is best performed using an agglomerative strategy [4]. Sequence-to-sequence distances are computed for all pairs of sequences, and the pairs for which the distance is below a threshold are merged. Merging is achieved by concatenating the original sequences, and recomputing the PCA subspace for the merged set. The process is repeated until the smallest distance exceeds a predefined threshold, or until a maximum number of iterations has been exceeded. This strategy means that the subspace defined by each cluster becomes more expressive as new, less similar clusters are merged. It does, however, have the disadvantage that if clusters representing two different people are merged, the combined cluster then represents both. The thresholds are therefore set conservatively, so that the data is "underclustered", with several clusters representing each character.

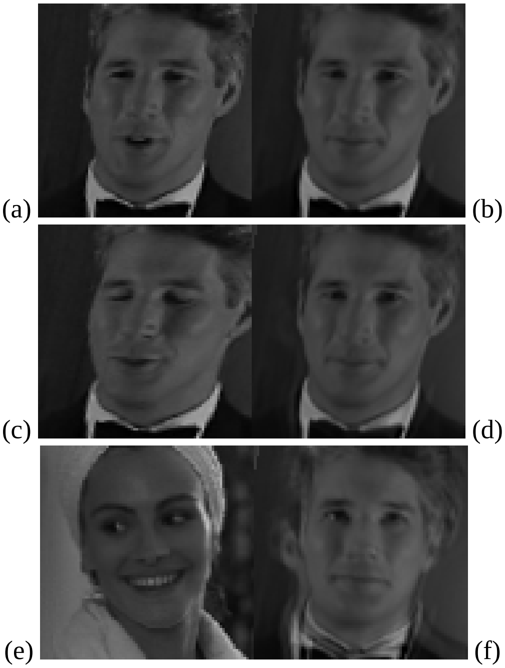

Figure 4: Projection onto a 5D subspace computed by automatic registration and principal components analysis. (a) Original image. (b) Projection removes the smile from the actor's face. (c) Original image. (d) Projection corrects for viewpoint-related distortion. (e),(f) Another face, and its projection.

Timing: All of the proposed distances take time of the order of 100ms (5sec with finite-difference derivatives) to compute in Matlab, so there is a clear advantage to sequence-to-sequence matching over multiple image-to-image matches. Given two average-length sequences, determining the match score by image-to-image matching is roughly 400 times more expensive than a single sequence-to-sequence match. For clustering, the speed improve-

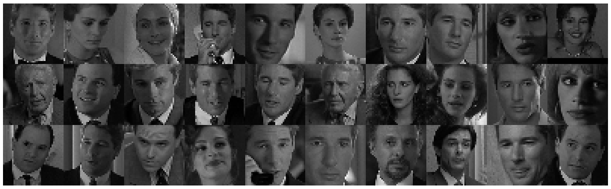

Figure 5: First images of clustered sequences. The main cast members are present, but duplicates have not been entirely suppressed. This remains a hard problem.

ments can be more dramatic. If using a technique which requires all pairwise distances, then a movie with 5000 faces requires roughly 12×10^6 image-to-image comparisons. The same movie arranged as 300 sequences requires $45,000$ sequence-to-sequence matches.

5. Conclusions

Matching images while remaining invariant to irrelevant changes such as lighting and viewpoint is a challenging problem. This paper has considered the case where the images are naturally segmented into slowly varying sequences, and the definition of invariant matching measures for such sequences. By representing each sequence as a low-dimensional space of images, the matching problem is reduced to finding the distance between subspaces under arbitrary, unknown, parametrized transformations. We introduced the joint manifold distance to solve this problem, and observe that an industrial-strength nonlinear optimizer will provide superior performance to the Newton methods commonly used in the literature.

The joint manifold distance for matching image sets is a superior approach to matching images independently for two reasons: first, pragmatically, fewer matches need be performed; second, computing the nearest point to a span of a linear space allows interpolation (i.e. effectively additional data is generated) over the image set, and this is where the gain is compared to multiple individual nearest neighbour matches.

Acknowledgements

We are grateful for discussions with Bill Triggs, David Capel and Michael Black. Krystian Mikolajczyk and Cordelia Schmid of INRIA Grenoble provided the face detection software [8]. Funding was provided by the Royal Society and the EC CogViSys project.

References

[1] Y. Adini, Y. Moses, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. *IEEE PAMI*, 19(7):721–732, 1997.

[2] R. Basri and D. Jacobs. Lambertian reflectance and linear subspaces. In *Proc. ICCV*, pages 383–390, 2001.

[3] A. W. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *Proc. ECCV*. Springer-Verlag, 2002.

[4] A. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, NJ, 1988.

[5] N. Jojic, B. Frey, P. Simard, and D. Heckerman. Learning mixtures of smooth, nonuniform deformation models for probabilistic image matching. In *Proceedings, AISTATS*, 2001.

[6] D. Keysers, J. Dahmen, T. Theiner, and H. Ney. Experiments with an extended tangent distance. In *Proc. ICPR*, pages 38–42, 2000.

[7] F. De la Torre and M. Black. Robust principal component analysis for computer vision. In *Proc. ICCV*, pages 362–369, 2001.

[8] K. Mikolajczyk, R. Choudhury, and C. Schmid. Face detection in a video sequence—a temporal approach. In *Proc. CVPR*, pages 96–101, 2001.

[9] J. J. Moré and D. C. Sorensen. Computing a trust region step. *SIAM J. Sci. and Stat. Comp.*, 3:553–572, 1983.

[10] H. Schneiderman and T. Kanade. A histogram-based method for detection of faces and cars. In *Proc. ICIP*, volume 3, pages 504 – 507, September 2000.

[11] H. Schwenk and M. Milgram. Constraint tangent distance for online character recognition. In *Proc. ICPR*, pages D:515–519, 1996.

[12] G. Shakhnarovich, J. W. Fisher, and T. Darrell. Face recognition from long-term observations. In *Proc. ECCV*, 2002.

[13] A. Shashua. On photometric issues in 3d visual recognition from a single 2d image. *IJCV*, 21:99–122, 1997.

[14] A. Shashua, A. Levin, and S. Avidan. Manifold pursuit: A new approach to appearance based recognition. In *Proc. ICPR*, 2002.

[15] P. Simard, Y. Le Cun, J. Denker, and B. Victorri. Transformation invariance in pattern recognition—tangent distance and tangent propagation. In *Lecture Notes in Computer Science, Vol. 1524*, pages 239–274. Springer, 1998.

[16] N. Vasconcelos and A. Lippman. Multiresolution tangent distance for affine-invariant classification. In *Advances in Neural Info. Proc. Sys. (NIPS)*, volume 10, pages 843–849, 1998.

[17] A. L. Yuille, D. Snow, R. Epstein, and P. Belhumeur. Determining generative models for objects under varying illumination: Shape and albedo from multiple images using SVD and integrability. *IJCV*, 35(3):203–222, 1999.