This is the published version:


Available from Deakin Research Online:

http://hdl.handle.net/10536/DRO/DU:30026094

Reproduced with the kind permission of the copyright owner.

Copyright: 2000, IEEE Xplore.
A Morphing Technique for Facial Image Representation

A.Z. Kouzani, S. Nahavandi, N. Kouzani, L.X. Kong, and F.H. She
School of Engineering and Technology, Deakin University
Geelong, Victoria 3217, Australia

Abstract

This paper presents a method for representation of facial images. The proposed method consists of two modules: face-image matching and face-image morphing. In the first module, the correspondence between two images is calculated for all pixel locations. A novel area-based matching method is proposed that makes use of the concept of the fractal dimension, and develops a non-parametric local transform as a basis for establishing correspondence between two face images. In the second module, a mapping is performed for deformation of the source face image onto the target face image. This is done to map the pixels in the source face image to the location of their corresponding pixels in the target image.

1 Introduction

The comparison of input face images with known face images in the existing face recognition systems, is based on the use of distance measurements taken in the space which represents face images. Although the existing face recognition systems use fairly standard distance metrics such as correlation coefficients, the distinguishing factor between the various systems is the type of face representation employed. A suitable face representation method can improve the recognition accuracy.

Pixel-based representation is the simplest representation method that is used in several face recognition systems [1, 2]. A major problem of this representation is that it cannot directly be used to obtain a linear combination of two face images. This problem can be reduced by a method known as correspondence based representation which involves the alignment of the faces using distinct points. This is why a variety of correspondence-based representations are considered in some recent face recognition systems [3, 4, 5]. The differences between the various correspondence-based representations lie in the collection of distinct points utilised for the alignment and in the way they align face images. In the former, some correspondence-based representations [3] use only a collection of predefined standard feature points, such as the centres of the eyes, the tip of the nose, the corners of the mouth, and the outline, while other methods [4, 6] use pixel-based correspondence. In the later, some methods [3] use simple image operation such as translation or scaling, while other methods [4, 5] utilise local image deformations.

A pixel-based correspondence method is proposed in this paper for representation of face images for obtaining better image alignments. The proposed face representation method consists of two modules: face-image matching and face-image morphing. In the face-image matching module, the correspondence between the detected face image and a reference face image is calculated for every pixel location. There are two approaches which are used for face-image matching: coarse-to-fine gradient-based optical flow [7], and area-based matching [8]. A novel area-based matching method is proposed which uses the concept of fractal dimension and develops a non-parametric transform as a basis for establishing correspondence between two face images. In the face-image morphing module, a mapping is performed for locally deforming the detected face image onto the reference face image. The mapping is performed using a triangle mesh warping method.

2 Proposed Method

In this section, the proposed Pixel-Based Correspondence method (PBC) is given. The PBC consists of two modules: face-image matching and face-image morphing. In the first module, the computation of the correspondence between two face images is performed for every pixel location. In the second module, the input face image is locally deformed so that its pixels are mapped to the appropriate places as those of pixels in the reference image. In the following, these modules are described.

2.1 Face-Image Matching

Matching can be defined as the establishment of correspondence between various data sets. The matching
problem is also referred to as the correspondence problem. Although the correspondence problem for human face morphing is defined specifically as the matching of a subset of predefined points, in this work the face correspondence problem is cast into the more general case where the aim is to match all of the data points. This can reduce the possibility of having false matches for a few facial landmarks. In addition, using more points for morphing will improve the quality of the morphed image.

For the calculation of the 2D displacement field between pixels across two images, a variety of methods are used and others continue to appear. Some of these methods are: feature-based, intensity-based differential, and area-based methods. A good overview of these methods can be found in [9,8]. It has been demonstrated that non-parametric area-based methods produce higher numbers of matched pixels than other matching methods [10,11]. However, they are not perfect and they still produce some mismatches. In the following, the concept of fractal dimension is used in a novel non-parametric image matching method proposed by the author.

Algorithm 1 (Fractal Dimension of a 2D Image) [18] Consider that the image of size M x M is scaled down to a size s x s where M/2 ≥ s > 1 and s is an integer. Then, an estimate of r is defined as r = s/M. Also, consider the 2D image as in a 3D space with (x, y) denoting 2D position and the coordinate z denoting gray-level. The (x, y) space is partitioned into grids of size s x s. On each grid there is a column of boxes of size s x s x s', where s' denotes the height of the box. If the total number of gray levels is G then \[G/s'] = [M/s].\] Let the minimum and maximum gray levels of the image in the (i, j)th grid fall in box numbers k and l, respectively. Then \[n_r(i, j) = l - k + 1\] is the contribution of \(N_r\) in the (i, j)th grid. Taking contributions from all grids, \(N_r\) can be obtained from

\[
N_r = \sum_{i,j} n_r(i, j),
\]

in which \(N_r\) is counted for different values of r (i.e., different values of s). Then using the basic fractal dimension equation

\[
D = \frac{\log(N_r)}{\log(1/r)},
\]

the fractal dimension D can be estimated from the least square fit of \(\log(N_r)\) against \(\log(1/r)\). Let \(y = mx + c\) be the fitted straight line, where \(y\) denotes \(\log(N_r)\) and \(x\) denotes \(\log(1/r)\). Then error of fit \(E\) can be expressed as

\[
E = \sqrt{\frac{\sum_{i=1}^{n} \left(\frac{(mx_i + c - y_i)^2}{(1 + m^2)}\right)}{n}},
\]

where \(n\) is the number of data points in the plot. This error provides a measure of fit so that the lower the value of \(E\), the better is the fit.

The procedure described above is the method which is used to calculate the fractal dimension of an entire face image. However, this method cannot be applied to the calculation of the correspondence between two images. In order to to use the fractal dimension for the calculation of the correspondence between two images, the author proposes representing each image pixel by the fractal dimension of a particular region of the image that surrounds the pixel. The calculation of the fractal dimension can be done by taking the image region as a full image and applying Algorithm 1 to it. However, this approach would face the following shortcomings: (i) it is possible that two local image regions with different textures and optical differences have the same fractal dimension; (ii) the selection of the size of the region need to be formulated; (iii) the regions surrounding the pixels located near image borders become smaller than the regions representing other pixels. In the following, the author proposes a method which can handle these shortcomings. To handle the first shortcoming, it is proposed that the fractal dimensions of image regions of various sizes are calculated and averaged. Therefore, by considering the effects of different image regions in the calculation of the fractal dimension of a pixel, it will no longer be possible to obtain similar fractal-dimension values for two separate local image regions with different textures. The use of differing-sized regions also solves the second shortcoming since it is no longer a requirement to select the region size. To handle the third shortcoming, the image extension process is proposed. This is done by circularly shifting the image. Therefore, the size of the regions surrounding the pixels located near image borders can remain intact.

Algorithm 2 (Fractal-Dimension Image Matching) The pixel-based correspondence between a source image \(I_S\) and a target image \(I_T\) is calculated as follows.

1. The n x n face images \(I_S\) and \(I_T\) are extended by circularly shifting their contents towards different directions. This will create two new m x m images \(\tilde{I}_S\) and \(\tilde{I}_T\) whose sizes are calculated from

\[
m = \begin{cases} 
  n + [n/2] - 1 & \text{if } [n/2] \text{ is odd} \\
  n + [n/2] & \text{if } [n/2] \text{ is even}.
\end{cases}
\]
2. For each pixel \( P_{ij} \) in \( I_S \) and \( I_T \), the following operations are separately performed.

(a) The window size \( W \) is initialised as \( \delta \).

(b) The associated image region of size \( W \), with pixel \( P_{ij} \) being its centre, is extracted from \( I_S \) or \( I_T \).

(c) The fractal dimension \( D_{ij}^W \) of the extracted image is calculated.

(d) The window size \( W \) is increased by 2.

(e) If \( W \) is less than or equal to \( n/2 \), then a jump to Step 2b is executed.

(f) The fractal dimension associated with \( P_{ij} \) is calculated from

\[
D_{ij} = \frac{1}{N} \sum_{w} D_{ij}^W,
\]

where \( N \) is the number of windows.

3. \( P_{ij} \) is replaced with \( D_{ij} \) (obtained from \( I_S \)) in \( I_S \) and with \( D_{ij} \) (obtained from \( I_T \)) in \( I_T \). This creates two new matrices \( I_S^\prime \) and \( I_T^\prime \).

4. The normalised-cross-correlation (NCC) [10] is performed on \( I_S^\prime \) and \( I_T^\prime \).

5. The correspondence field is stored.

A validation technique adopted from [13] is used to identify incorrect matches. In this technique, the roles of the two images are reversed and the matching is performed a second time. Since the matching is performed on the same two images, the second correspondence field should be inverse of the first one. By comparing the two correspondence fields, the invalid matches can easily be detected and removed. Therefore, this validity test enhances the image-matching results.

2.1.1 Experimental Results

The proposed image-matching method together with a number of other matching metrics have been implemented, and their behaviours have been tested on a collection of face images. Ten pairs of face images of resolution 314 x 229 have been used in the experiments; each pair contains two face images of different people. The poses of the face images of each pair are approximately similar, with a possible pose variation of maximum ±5°. The ten pairs of face images are separated into two groups. Five pairs of face images, taken under similar lighting conditions, are put in the first group. The remaining pairs, in which the source face image is 30% brighter than the target image (see Figure 1 for an example), are put in the second group. Both the pose and the lighting conditions vary amongst the ten different pairs.

The existing methods used are the Bergen-Hingorani gradient-based optical flow (BH) [7], the NCC, the rank[10], and the census [11]. For the NCC and rank metrics, the correlation window size of 9 x 9 is used. For the census metric, a correlation window size of 9 x 9 and a transform window size of 7 x 7 are used. Other parameters, such as the number of disparities considered, are held constant for all methods. For each group of the test images, the remaining proportion of matches after each validity check are added and then divided by the number of pairs in that group, and are displayed in Table 1. The larger the proportion of matched pixels, the better is the performance of the matching system.

![Figure 1: A test pair in which the source image (a) is 30% brighter than the target image (b).](image)

| Table 1: Proportion of matched pixels for each correspondence method, for each group of test images. |
|--------|--------|--------|--------|--------|
| BH     | NCC    | rank   | census | Proposed |
| Group 1| 0.59   | 0.68   | 0.66   | 0.81    | 0.91    |
| Group 2| 0.32   | 0.42   | 0.54   | 0.77    | 0.86    |

2.1.2 Discussions

It can be seen from Table 1 that the Bergen-Hingorani method obtains the lowest matching rate amongst the tested methods. Amongst the NCC, the rank, and the census, the latter achieves the highest proportion of matched pixels for the two test groups. However, the use of the proposed image matching method allows even higher matching rates than those of the census for both test groups.

The results prove that the proposed image matching method is an efficient method for computing correspondence and it accounts for variation in the source and target images.
2.2 Face-Image Morphing

Image morphing is a technique for producing transitions between images [14, 15]. This technique is often used in the media industry to achieve special effects. Before the development of morphing, image transitions are achieved through the use of colour interpolation (i.e., cross-dissolve), for instance, linear interpolation to fade from one image to another. Figure 2 displays this process applied over 5 frames. The results are poor, owing to the double-exposure effects apparent in the middle frames, where both input images contribute equally to the output. Image morphing is the combination of image warping with a cross-dissolve between images. This process is explained below.

2.2.1 Image Warping

Warping is the act of distorting an image according to a mapping between a source space \((u, v)\) and a target space \((x, y)\) [16, 17]. The mapping is usually specified by the function \(x(u, v)\) and \(y(u, v)\). Image warping is used in image processing primarily for correction of geometric distortions introduced by imperfect imaging systems. Wolberg’s book [17] effectively covers the fundamentals of digital image warping, culminating in a mesh warping technique which uses spline mapping in two dimensions. This technique has the advantages that it is both fast and intuitive, and efficient algorithms exist for computing the mapping of each pixel from the control grid.

2.2.2 Cross-Dissolve

Cross-dissolve is a useful colour transformation. Given two images of \(f(x, y)\) and \(g(x, y)\), a cross-dissolve between them is a group of transformations of the image colour space (see Figure 2).

2.2.3 Morphing Techniques

Image morphing between two images begins with establishing their correspondence between pairs of feature primitives, e.g., mesh nodes, line segments, curves, or points. Each primitive specifies an image feature, or landmark. The feature correspondence is then used to compute the mapping functions that define the spatial relationship between all points in both images. Mapping functions interpolate the positions of the features across the morph sequence. Once both images have been warped into alignment for intermediate feature positions, ordinary cross-dissolve is performed to generate middle images.

Feature specification is the most tedious aspect of morphing. Although the choice of allowable primitives may vary, all morphing approaches require careful attention to the precise placement of primitives. Given feature correspondence constraints between both images, a warp function over the whole image plane must be derived. This process, referred to as warp generation, is essentially an interpolation problem. Another interesting problem in image morphing is transition control. If transition rates are allowed to vary locally across middle images, more interesting animations are possible. As the animation proceeds, the source image is smoothly faded out and distorted towards the alignment of features in the target image. The target image, which is initially distorted to the feature geometry of the source image, is faded in so that the distortion is gradually reduced. Therefore, the first images of the sequence resemble the source image, while the last images are similar to the target image. The middle image of the animation contains an average of the source and the target image adjusted to an average feature geometry.

There are a variety of morphing methods in the literature [15]. Each of these methods has its own advantages and disadvantages. The morphing method which is used in this work because of its relative simplicity and higher speed, is a branch of mesh warping.

2.2.4 Triangle Mesh Warping

Triangle mesh warping consists of dissecting the domain space into a suitable set of triangles with the given data points being the corners of the triangles. The source im-
image is referred to as $I_S$ and the target image is referred to as $I_T$. The source image has an associated mesh $M_S$ that specifies the coordinates of control points, or landmarks. A second mesh, $M_T$, specifies their corresponding positions in the target image. Face landmarks such as the eyes, nose, and lips should lie below corresponding grid lines in both meshes. Together, $M_S$ and $M_T$ are used to define the spatial transformation that maps all points in $I_S$ onto $I_T$. The meshes are constrained to be topologically equivalent, i.e., no folding or discontinuities are permitted. Therefore, the nodes in $M_T$ may wander as far from $M_S$ as necessary, as long as they do not cause self-intersection.

Figure 3: Triangulation of a face image using 80 distinct points. a) Voronoi diagram. b) Delaunay triangulation.

2.2.5 Triangulation

Triangulation of a set of points is a process that is done in computer graphics [18]. Several different triangulation techniques can be used for triangulation of a set of points. An optimal solution is to carry out a triangulation so that the points inside a triangle are closer to its own vertices than to the vertices of any other triangle. Delaunay triangulation [19] is a popular method for optimal triangulation of a set of points. To calculate the Delaunay triangulation, Voronoi regions [19] should be obtained. Many different direct and indirect construction methods are available for the Delaunay triangulation. The method that is adopted in this work is the combination of the quality triangular mesh generation proposed by Ruppert [20] and the divide-and-conquer incremental Delaunay triangulation proposed by Guibas and Stolfi [19]. These two triangulation techniques have been reported to generate more accurate results than the existing counterparts and to be faster than them. Our method works by local modification of the diagram to allow the insertion of a new vertex. The image can be seen as a driver for the partition. It guides the evolution and location of triangles. All triangles are recorded in a graph structure. A triangle is coded by its address (position) in the graph. Voronoi diagram and Delaunay triangulation of a sample face image is shown in Figure 3. A set of 80 distinct points are used for computation of the Delaunay triangulation. Having described the morphing method, an example of image morphing is displayed in Figures 4.

Figure 4: Morphing between two face images.

3 Concluding Remarks

In this paper, a pixel-based correspondence method is presented for representation of face images. The proposed method improves the performances of the existing image representation methods because: (i) it aligns face images, (ii) the face alignment is performed for each pixel location, (iii) it employs the proposed image matching method to establish the correspondence between two face images, and (iv) it utilises the image morphing methods which allow local deformations of the face images in the alignment process. The pro-
Acknowledgments

The first author would like to thank DEETYA and FUSA for their financial supports.

References


