Image Warping by Radial Basis Functions: Application to Facial Expressions

Nur Arad*, Nira Dyn, Daniel Reisfeld†, Yehezkel Yeshurun
School of Mathematical Sciences
Tel-Aviv University
Tel-Aviv 69978, Israel.

Abstract

The human face is an elastic object. A natural paradigm for representing facial expressions is to form a complete 3D model of facial muscles and tissues. However, determining the actual parameter values for synthesizing and animating facial expressions is tedious; evaluating these parameters for facial expression analysis out of grey-level images is ahead of the state of the art in computer vision. Using only 2D face images and a small number of anchor points, we show that the method of radial basis functions provides a powerful mechanism for processing facial expressions. Although constructed specifically for facial expressions, our method is applicable to other elastic objects as well.

1 Introduction

Current general object recognition schemes in computer vision fail to recognize human faces since the face is an elastic object that is subject to major spatial deformations due to the modification of facial expressions. Yet, humans can recognize faces even under the most extreme spatial variations caused by these expressions.

In order to be able to recognize faces by computers, face images must be analyzed and then normalized, namely, be transformed to some invariant representation. This ability to animate facial expressions has been attracting attention in computer graphics

*e-mail: nur@math.tau.ac.il
†e-mail:reisfeld@math.tau.ac.il

as well, and has many applications such as dynamic facsimile, low bandwidth video and teleconferencing. Such applications require realistic reproduction of faces, as opposed to computer vision face recognition and classification tasks that do not require full reconstruction.

Current approaches for synthesizing facial expressions use 3D face models, which employ a 3D mesh. Some try to simulate muscle action, skin complexion and so on [30, 19, 24, 28], while others employ texture mapping techniques, which transform 2D texture planes onto the manifold determined by the 3D mesh representing face geometry [11, 16, 32, 31, 21]. The two methods can be combined to enhance their respective performance. However, determining the actual parameter values for synthesizing and animating facial expressions is a task requiring tedious interactions with a human operator. In the case of facial expression analysis, the situation is even worse. Evaluating the model parameters out of grey-level images accurately enough is ahead of the state of the art in computer vision. An alternative approach is not to rely on a 3D model of the face, but rather to use a limited set of anchor points in 2D face images. This approach is attractive from the computational complexity point of view and is supported by psychophysical findings [4]. We have recently demonstrated the applicability of such an approach to face recognition. We have developed a system that automatically detects the most important facial features (eyes and mouth) using *generalized symmetry* [25, 26]. We have also shown that normalizing a 2D image of a face using an affine transformation determined by the location of the eyes and mouth is an effective step towards face recognition [10]. The affine transformation can compensate for various viewing conditions, but is not effective if the facial expression is modified. A recent 2D approach [1], which was impressively used in a video clip, ignores the affine aspects of the transformation.

All these points have motivated us to look for a smooth 2D transformation, which can be used to compensate for changes in facial expressions, based on a relatively small number of anchor points. A 2D transformation should be robust in the sense that the anchor points need not be specified very accurately. Moreover, after considering global constraints, the position of each anchor point should have a local effect, reflecting the elasticity of the face and the relative independence of its parts.

In the following we show that the theory of radial basis functions provides a powerful mechanism for image warping and demonstrate its application to face images.

2 The Mapping

We regard images as 2D objects. In this respect, an image is a finite domain of a plane with a grey level (or color) associated with each point. A warping of an image is then primarily a transformation of the plane to itself, and the grey level values are transformed according to the transformation of their associated coordinates. Our main concern is the construction of a mapping of images (planes) that is determined by the mapping of a small number of *anchor points* – points whose mapping is predetermined. This requirement leads us to interpolation theory.

2.1 Radial Basis Functions

Radial functions have proven to be an effective tool in multivariate interpolation problems of scattered data: Given a univariate function $g : \mathbb{R}^+ \to \mathbb{R}$ one may attempt to interpolate the scattered d-dimensional data

$$(\bar{x}^i, F_i) \quad , \quad \bar{x}^i \in \mathbb{R}^d \quad , \quad F_i \in \mathbb{R} \quad , \quad i = 1, 2, \ldots, N,$$

by a *radial function* ,$S(\bar{x})$, represented by

$$S(\bar{x}) = \sum_{i=1}^{N} a_i g\left(\| \bar{x} - \bar{x}^i \|\right), \tag{1}$$

where $\| \cdot \|$ denotes the usual Euclidean norm on $\mathbb{R}^d$. A function of this type is usually referred to as a *pure radial sum*. The choice of a radial function reflects the fact that the scattered data has no preferred orientation, and the fact that for given i the data point $\bar{x}^i$ equally effects all points of equal distance to $\bar{x}^i$. Interpolation by sums of the form (1) is possible whenever the system of linear equations

$$\begin{gathered} G\bar{a} = F \quad , \quad G = g_{ij} = g\left(\| \bar{x}^i - \bar{x}^j \|\right), \\ \bar{a} = (a_1, a_2, \ldots, a_N)^T, F = (F_1, F_2, \ldots, F_N)^T \end{gathered} \tag{2}$$

has a unique solution. Some classes of functions for which a unique solution to (2) exists for any N distinct points $\bar{x}^i \in \mathbb{R}^d$, and are well known in the literature [7] are:

1. $g(t) = (t^2 + c^2)^\alpha \quad , \quad 0 < \alpha < 1$ (multiquadrics).
2. $g(t) = \log(t^2 + c^2)^{1/2} \quad , \quad c^2 \geq 1$ (shifted log).
3. $g(t) = \exp(-t^2/\sigma^2) \quad , \quad \sigma > 0$ (Gaussian).

The drawback of using pure radial sums for our purposes lies in the fact that these sums do not reproduce polynomials, and thus yield a poor approximation of the transformation for points far away from the data points $\bar{x}^i$. In particular, the natural transformation determined by three anchor points in general position in the plane is the affine mapping, however this mapping cannot be realized by pure radial sums. Thus for our application we use interpolants of the form

$$S(\bar{x}) = \sum_{i=1}^{N} a_i g(\| \bar{x} - \bar{x}^i \|) + p_m(\bar{x}) \quad , \quad p_m(\bar{x}) \in \Pi_m \quad , \tag{3}$$

satisfying

$$S(\bar{x}^i) = F^i \quad , \quad i = 1, 2, \ldots, N \quad , \quad \sum_{i=1}^{N} a_i q(\bar{x}^i) = 0 \quad , \quad \forall q \in \Pi_m \tag{4}$$

Here Π_m is the space of all algebraic polynomials of degree at most m on $\mathbb{R}^d$. This method of interpolation reproduces polynomials in Π_m whenever (4) is uniquely solvable. Conditions for the unique solvability of (4) can be found in [20].

A large class of radial functions for which (4) is solvable at distinct $\{\bar{x}^i\}$ has the additional property that the interpolant satisfies some variational principle, namely the interpolant minimizes some functional defined on a relevant space of functions. For example, the interpolant (1) with g the Gaussian radial basis function (normalized with $\sigma = 1$, and without a polynomial term) minimizes the functional (f, f), where (f, h) is defined by

$$(f, h) = \iint_{\mathbb{R}^2} \hat{f}(\lambda) \cdot \overline{\hat{h}(\lambda)} \cdot \exp(\lambda^2) d\lambda,$$

($\hat{f}$ is the fourier transform of f), and the minimum is taken over all functions f for which the functional (f, f) is defined [18]. The relevance of such a variational principle for our applications seems quite remote.

2.2 Designing the Mapping

Since we are interested in $\mathbb{R}^2 \to \mathbb{R}^2$ mappings, and interpolation deals with $\mathbb{R}^d \to \mathbb{R}$ functions, we construct our mapping by using a pair of functions $\mathbb{R}^2 \to \mathbb{R}$. Given two 2-dimensional data sets (x^i, y^i) and (u^i, v^i) $i = 1, 2, \ldots, N$ (anchor points) we are looking for a transformation $T = (T_U, T_V) : \mathbb{R}^2 \to \mathbb{R}^2$ with the following properties:

(a) T is a radial function in each of its components. This reflects the requirement that the effect of each anchor point, $\bar{x}$, is the same for all equidistant points from $\bar{x}$.

(b) $T(x^i, y^i) = (u^i, v^i)$ for all $i = 1, 2, \ldots, N$ (interpolation).

(c) The number of data points to be interpolated is any number $N \geq 3$; if the number of anchor points is 3, the mapping is affine, and an arbitrary number of additional anchor points may be used.

(d) The components of T reproduce linear polynomials on $\mathbb{R}^2$. This condition guarantees that T will be an affine transformation whenever the interpolation data admits such a transformation.

(e) There will be a tradeoff between the *warping condition* – warping the plane as little as possible, and the *locality condition* – the interpolation of the anchor points will have a local effect.

(f) In some cases the anchor point mapping needs not be exact. In these cases we would like a tradeoff between the interpolation error (of the anchor points) and the minimal warping condition stated in the previous condition (**e**).

Conditions (**a**-**d**) are satisfied by $T = (T_U, T_V)$, where T_U and T_V are radial functions of the type (3) with $m = 1$. Condition (**e**) needs some mathematical formulation. We postpone the discussion on locality, and concentrate for the time being on the warping condition. Given a function $f : \mathbb{R}^2 \to \mathbb{R}$ which is twice continuously differentiable, it is customary to use the functional

$$J(f) = \iint_{\mathbb{R}^2} \left[(f_{xx})^2 + 2(f_{xy})^2 + (f_{yy})^2 \right] d(x, y)$$

as a measure of the total amount of bending of the surface $(x, y, f(x, y))$ [6]. The functional J is rotation invariant, again reflecting the fact that the data has no preferred orientations. We note that the functional J is only an approximation to the total bending energy of the surface $(x, y, (f(x, y))$, however the 2D surface minimizing J and interpolating the data $\{(x_i, y_i, z_i)\}_{i=1}^N$ is referred to as a *thin-plate spline* and is known as a good approximation to a thin steel plate under stress.

Recall that our mapping is defined for each coordinate separately, therefore we are looking for a transformation $T = (T_U(x, y), T_V(x, y))$ such that

$$T_U \in \left\{ f \;\middle|\; f(x^i, y^i) = u^i \;\;,\;\; i = 1, 2, \ldots, N \right\}$$

$$T_V \in \left\{ f \;\middle|\; f(x^i, y^i) = v^i \;\;,\;\; i = 1, 2, \ldots, N \right\}$$

$$J(T_U) + J(T_V) \text{ is minimal.}$$

This is another approximation to the actual underlying variational problem, namely the minimization of the total warping induced by the mapping. Minimizing $J(T_U) + J(T_V)$ can be performed by the separate minimization of $J(T_U)$ and $J(T_V)$.

With this formulation in mind, it is known that the choice $g(t) = t^2 \log t$ (with $g(0) = 0$) provides a uniquely solvable interpolation problem (3) – (4) with $m = 1$, the solution of which minimizes the functional J [6]. Thus the transformation $T = (T_U, T_V)$ will be of the form

$$T(x, y) = \left(\alpha_1 + \alpha_2 x + \alpha_3 y + \sum_{i=1}^{N} a_i g_i(x, y) \;\;,\;\; \beta_1 + \beta_2 x + \beta_3 y + \sum_{i=1}^{N} b_i g_i(x, y) \right) \qquad (5)$$

with $g_i(x, y) = \| (x - x^i, y - y^i) \|^2 \cdot \log(\| (x - x^i, y - y^i) \|)$. The computation of the coefficients in (5) involves the solution of two square linear systems of size $N + 3$ (with the same matrix in each case). An algebraic treatment of the mapping (5) is given in [2].

As stated in property (**f**), we are willing in some cases to relax the interpolating conditions $T(x^i, y^i) = (u^i, v^i), i = 1, 2, \ldots, N$ and in turn we wish to further reduce the bending factor $(J(T_U), J(T_V))$ of the transformation T. In such a case we are to find twice differential functions T_U and T_V such that for a given $\lambda > 0$ the functionals

$$\tilde{J}(T_U) = \sum_{i=1}^{N} \left[u^i - T_U(x^i, y^i) \right]^2 + \lambda J(T_U) \tag{6}$$

and

$$\tilde{J}(T_V) = \sum_{i=1}^{N} \left[v^i - T_V(x^i, y^i) \right]^2 + \lambda J(T_V) \tag{7}$$

are minimized. The solution of this variational problem for T_U is again given by

$$T_U(x, y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \sum_{i=1}^{N} a_i g \left(\| \; x - x^i \; \| \right),$$

the coefficients $\alpha_1, \alpha_2, \alpha_3, a_1, a_2, \ldots a_N$ are solutions of the linear system

$$(G + \lambda I)_i (a_1, \ldots a_N)^T + \alpha_1 + \alpha_2 x_i + \alpha_3 y_i = u_i \quad \text{for} \quad i = 1, \ldots, N \tag{8}$$

and

$$\sum_{i=1}^{N} a_i q(\bar{x}^i) = 0 \quad \text{for} \quad q(x, y) = 1, x, y, \tag{9}$$

where G is defined in equation (2), and $(G + \lambda I)_i$ is the i'th row of the matrix $G + \lambda I$ [9]. A similar solution exists for T_V. The equations given in (8) are the generalization of the interpolation equations, while those given in (9) guarantee the reproduction of linear polynomials.

Some special cases of the functionals (6)–(7) are listed below:

- $N = 3$. The minimum bending transformation of the plane reduces to an affine transformation (for which $J = 0$). Thus we meet requirement (**c**).

- $\lambda = 0$. Minimization of $\tilde{J}$ results in an exact interpolant which minimizes J. Thus we prefer exact mapping of the anchor points rather than a low bending factor.

- $\lambda \to \infty$. Minimization of $\tilde{J}$ yields the affine mapping that minimizes the sum of the squares of distances from $T(x^i, y^i)$ to $(u_i, v_i), i = 1, ..., N$. Notice that

$$\sum_{i=1}^{N} \left[v^i - T_V(x^i, y^i) \right]^2 + \sum_{i=1}^{N} \left[u^i - T_U(x^i, y^i) \right]^2 = \sum_{i=1}^{N} \left\| T(x^i, y^i) - (u_i, v_i) \right\|^2$$

- $0 < \lambda < \infty$. The parameter λ controls the tradeoff between the affine transformation that annuls the bending factor J and the radial factor that interpolates the data, as stated in property (**f**). In this formulation, the same value of λ is chosen for all points. We note, however, that in a general setting, different values of λ can be chosen for different points.

Returning to property (**e**), we at last turn to the locality condition. The functional J has a global nature, thus a small perturbation of one of the anchor points effects all points in the transformed plane. In some cases we prefer some or all of the anchor points to have only a local effect. To this end we will sometimes switch from the basis function $r^2 \log r$ to a radial basis function which incorporates a *locality parameter*, such as the Gaussian radial basis function $g(r) = \exp(-r^2/\sigma^2)$, $\sigma > 0$,where the parameter σ may be used to control the locality of each radial function: As we increase σ, the effect of the radial part on the interpolant is more global to the anchor points, while the total bending energy is decreased. Each anchor point may have a distinct value of the locality parameter. Determining this parameter is application dependent, while the choice of the thin plate basis function leaves no free parameters.

A practical feature of the transformation is invariance of parameter values to image size. Therefor, λ of equations (6 – 8) is normalized such that $\lambda := (10^{-4}wh)^{3/2}N\lambda$, where w and h are the image width and image height given in pixels. By the same token, Gaussians are adjusted to the image size such that 1σ correspond to $(h+w)/12$.

3 Applications

We have constructed a continuous mapping for discrete pictures given in pixels. Its implementation involves overcoming aliasing problems. A standard procedure is to apply the *backward transformation* – the inverse transform from target to source. However, if more then three anchor points are used, the *forward transformation* – the source to target transformation, is not the inverse of the backward transformation. This drawback notwithstanding, in most practical cases backward transformations can be safely used. In cases where two anchor points are mapped to the same or almost the same location, the backward transformation is ill-defined and the forward transformation must be used. Overcoming the resulting aliasing problem is also a standard procedure [16]. We have found that, in practice, the two directions are useful. Of the following examples Figures 6 and 7 were obtained using the forward transformation, and the rest were obtained using the backward transformation. When using backward transformations, bilinear interpolation was used for antialiasing.

3.1 Determining mapping parameters

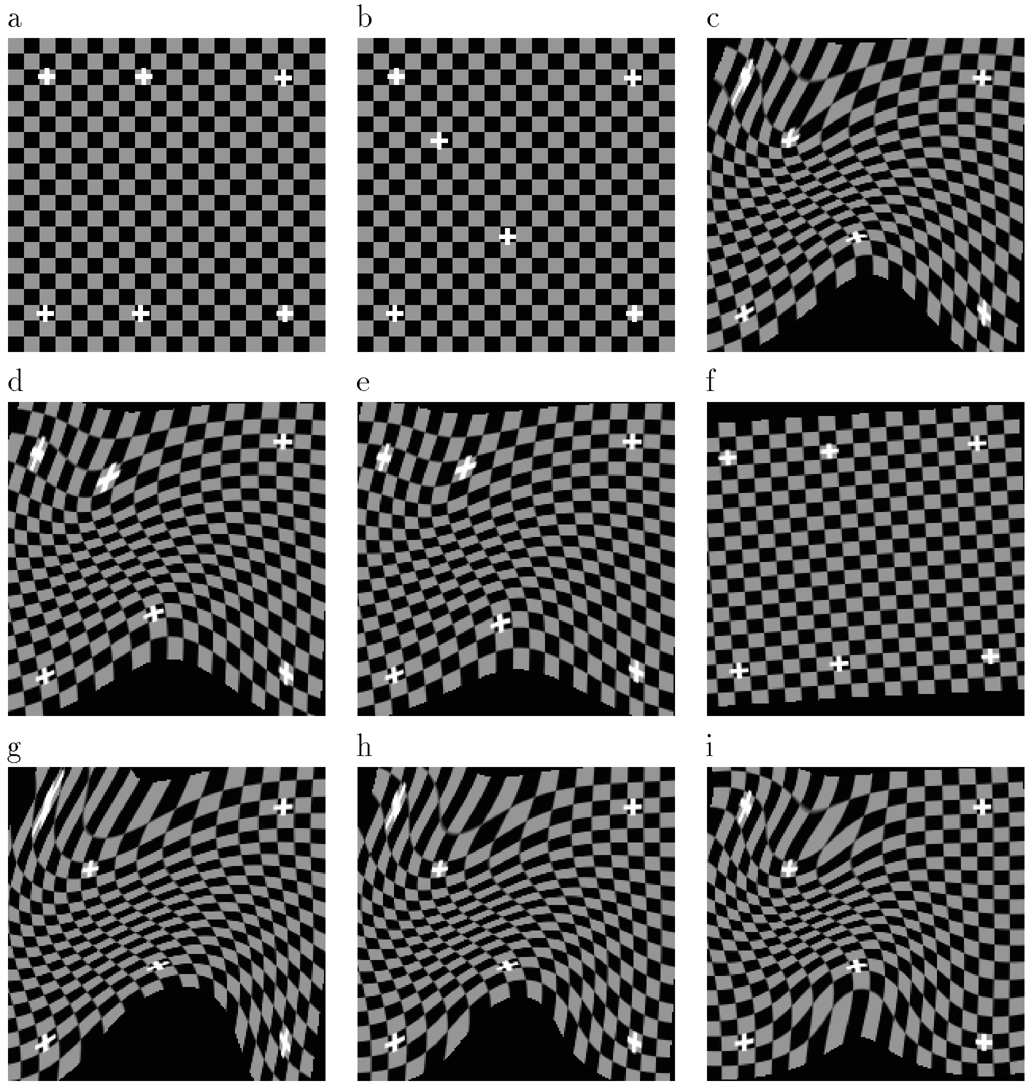

Figure 1: Warping using various radials and other control parameters. (**a**) Source. (**b**) Destination. (**c**) – (**f**) Thin plate spline radial basis with $\lambda = 0,\ 0.1,\ 10,\ 100$ respectively. (**g**) – (**i**) Gaussian radial basis with $\lambda = 0$ and $\sigma = 2.5, 2, 1.5$ respectively.

The free parameters of the family of mappings defined above are the radial function, its locality parameter, σ, when the thin-plate is not used, and the tradeoff parameter, λ, between the warping condition and the locality condition. In figure 1 a checkerboard is warped using six anchor points. Four of the points are fixed and the other two are moderately shifted (**a**) and (**b**). The first four warps use the thin plate basis function with different tradeoff between the interpolation error and minimal warping – λ of equations (6) and (7). (**c**) is pure thin plate interpolation, (**f**) is almost the affine transformation that minimizes the sum of the squares of the distances between the desired transformation of the anchor points and their actual transformation. (**d**) and (**e**) are intermediate cases. The last three mappings of Figure 1 demonstrate the tradeoff between the warping condition and the locality condition using Gaussian radials. Notice that when the Gaussian become narrower (σ increases), the warping is more local and of a wilder nature close to the anchor points while almost unnoticeable far away from these points.

3.2 Generalization of affine mappings

The use of similarity transformations for face normalization dates back to 1878 when Sir Francis Galton devised a photographic technique called composite photography, in which he superimposed images of two or more faces by means of multiple exposures. For the technique to succeed, he carefully aligned the different images so that the pupils of the eyes coincided. He superimposed photographs of faces of army personnel for a definite portrait of health; of tuberculosis victims for disease; and of convicted felons for criminality [14, 15]. Being a member of the Victorian elite, he was surprised to see that a superimposed photograph of people convicted of murder, manslaughter, or violent robbery tended to look more respectable than the individual ones used to make it. A straightforward generalization of the similarity transform, determined by two points, is the general affine transformation determined by the position of three anchor points. Using the center of the mouth as the third anchor point improves the quality of the superposition of facial images and is instrumental for face recognition [10].

Figure 2 demonstrates Galton's method and its improvement. Instead of superimposing photographs by chemical means, a similarity transformation (translation, rotation and scaling) is used. A generalization is obtained by using an affine transformation (similarity and shear).

Notice that the matching at the chin leaves room for improvement. This is fine for recognition purposes, since Mona and Venus posses different types of chins. Assume, however, that one wants to change Venus' chin to one that is similar to Mona's. This is possible by simply adding another anchor point on the chin and using the thin-plate spline as demonstrated in figure 3. Notice that apart from the chin area the two transformations are not easily distingushable.

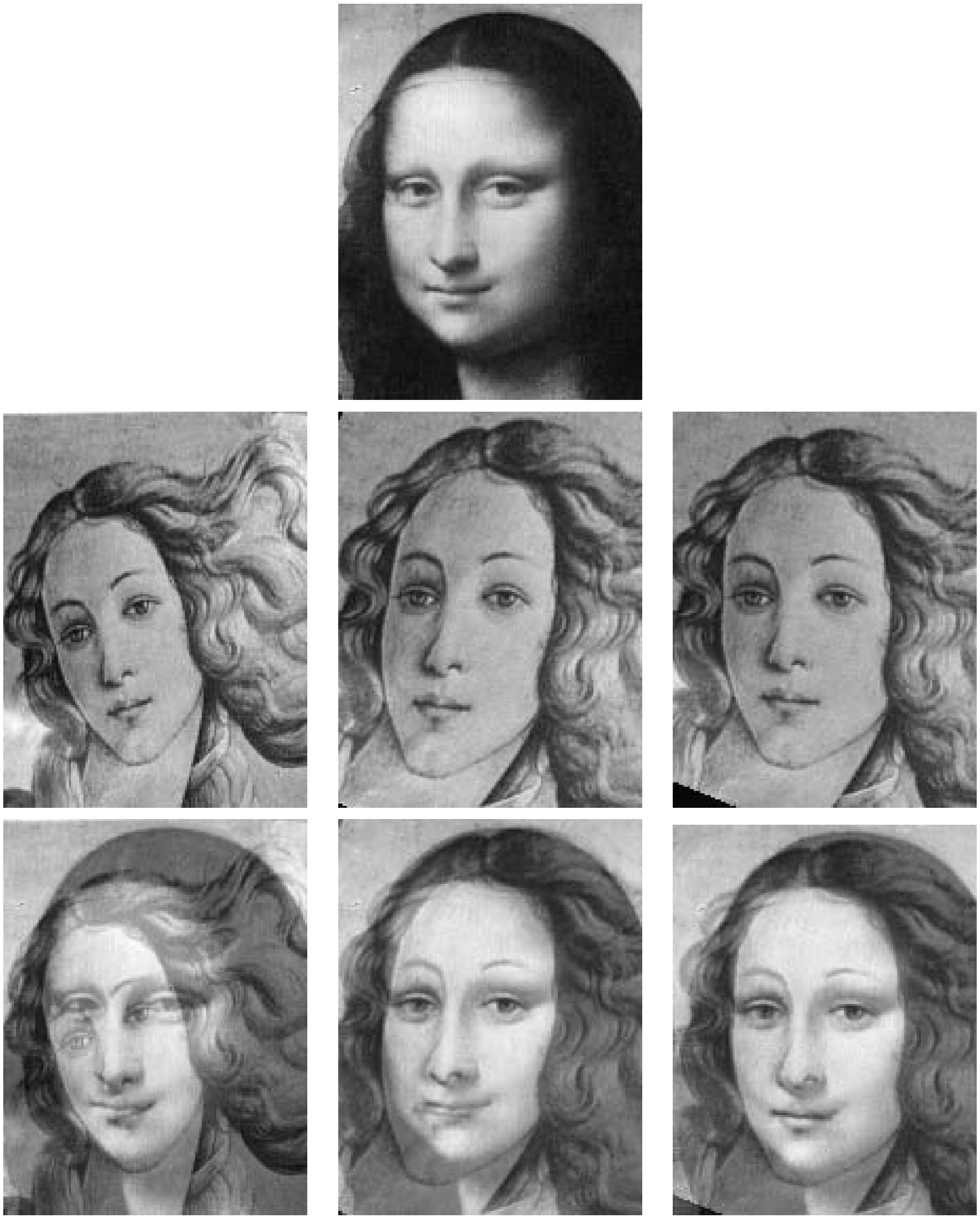

Figure 2: Alignment by eyes and mouth. **Top**: original Monalisa. **Middle** (left to right): original Venus, Venus aligned to Monalisa by the similarity transformation determined by the eye's location, Venus aligned to Monalisa by the affine transformation determined by the eyes and mouth. **Bottom**: Superposition of Monalisa with each of the Venus images above.

Figure 3: **Left:** Warping of Venus using four points and the thin-plate spline **Right:** Superposition of the left image with the Mona Lisa.

3.3 Animation and facial expressions

The first motivation behind this research is to develop an effective facial expression transformation using a small number of anchor points.

Three anchor points are effective in overcoming distortions due to camera positions and some other mild distortions. However, one might expect that more points are needed for dealing with expressions. It is surprising to see that in the interpolation of 2D images, the use of a small number of points is sufficient in many cases. Figure 4 demonstrates the effect of using six carefully chosen anchor points. Figure 5 demonstrates that drastic changes in expression can be obtained by a subtle use of more points.

The technique is useful for animation sequences as well. Figure 6 shows frames from a videotape in which a single image was used to produce a whole live sequence. The tradeoff parameter, λ, used in equations (6) and (7) is used to control continuous changes in facial expressions. The first image in the sequence is the original, the last is the warped original using exact interpolation with Gaussian radials. Intermediates were produced using decreasing values of the tradeoff parameter λ.

The Gaussian radials, which enable localization, are used in figure 6, and the animated face is realistic. Note that if minimal warping is pursued, the thin plate radials should be used and the result is shown in Figure 7. Comparing this poor result with that of the previous figure, it is rather evident that the use of the minimal warping functional is not always adequate for warping purposes.

4 Real-time implementation

We have implemented the mapping with run-time that is comparable with time needed to load an image and present it on the screen. In order to achieve this fast implementation, some special care should be taken at certain points.

Let N be the number of anchor points. The coeficients of the mapping are the solution of an $(N+3) \times (N+3)$ linear system of equations. Since N is much smaller than the image size, any standard method solves the system in neglegable time. Therefore, we concentrate on the mapping itself.

At this point we distinguish between thin-plate splines and basis functions with local influence (e.g. Gaussians). In both cases one can use a look-up table instead of computing the function values. In the case of local influence, function values can be approximated by zero for large arguments, and a small table can store the values for small arguments. For example, we store Gaussians in a table of size 3σ. Thin-plate splines do not decay for large arguments, and their look-up table may turn out to be too large for some configurations. Nevertheless, since $\log(2^k M) = k\log(2) + \log(M)$ for any M, a table of size M can be used together with shift operations. This implementation reduces memory requirements while increasing the run-time by a few shift operations

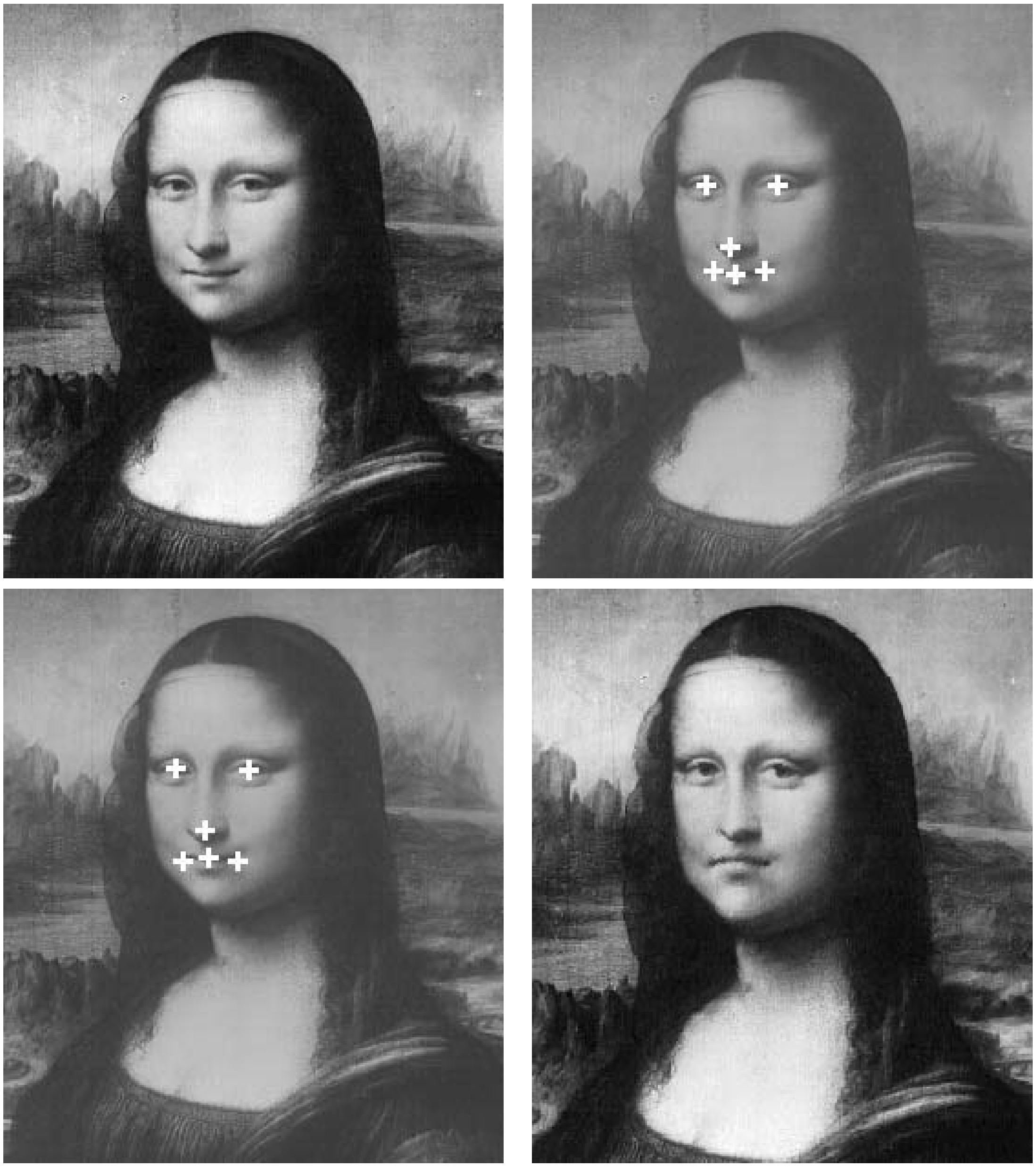

Figure 4: The Smile lost using only six points and the thin plate radial. **Top**: Original image. and source of the anchor points marked by crosses. **Bottom**: Destination of the anchor points and warped image.

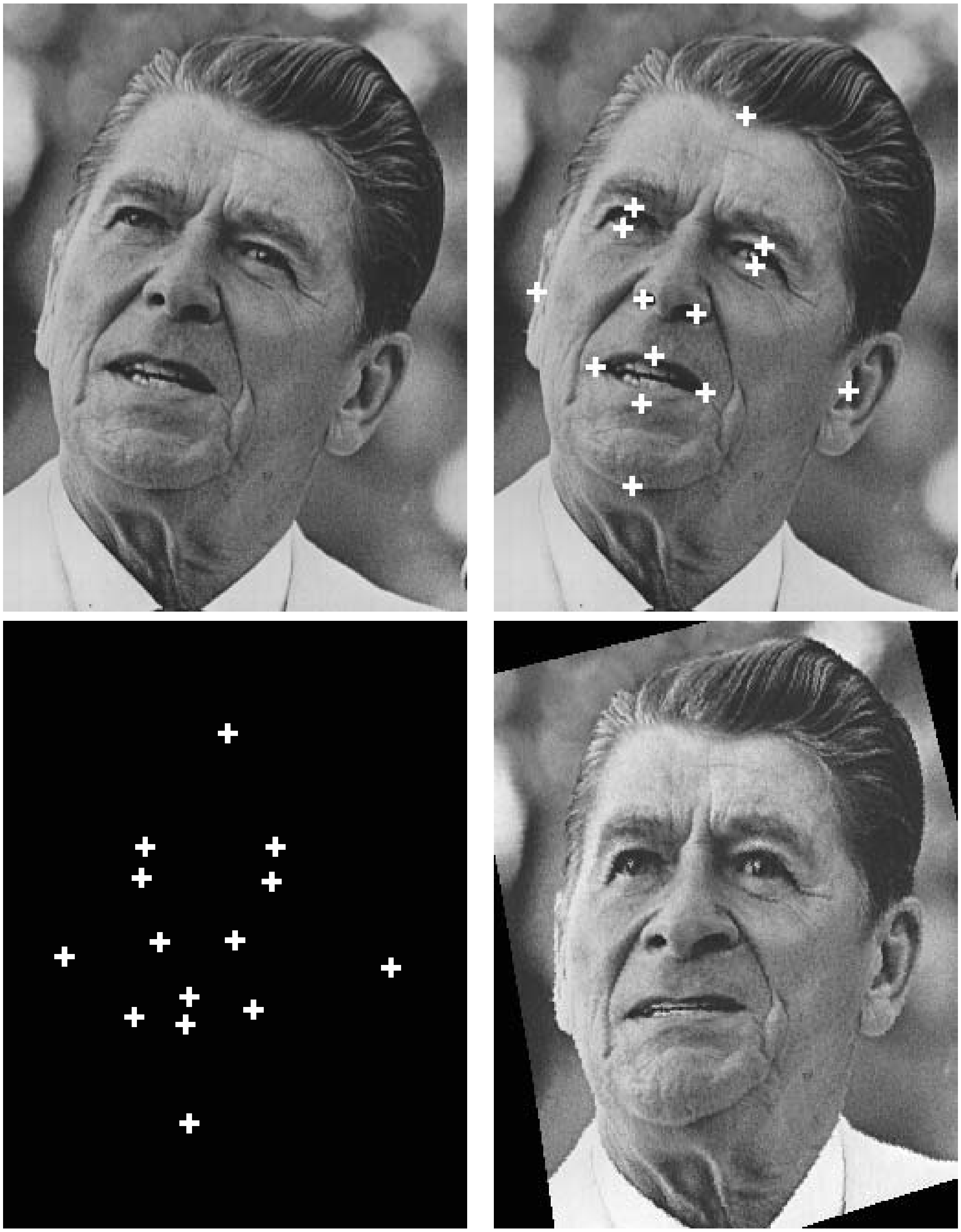

Figure 5: Change in expression using 14 points. **Top**: Original image and anchor point locations. **Bottom**: Anchor point destinations and warped image. Thin plate radials were used. Notice the cumulative effect of the minor changes in the positions of the anchor points in the eyes, right nostril and mouth.

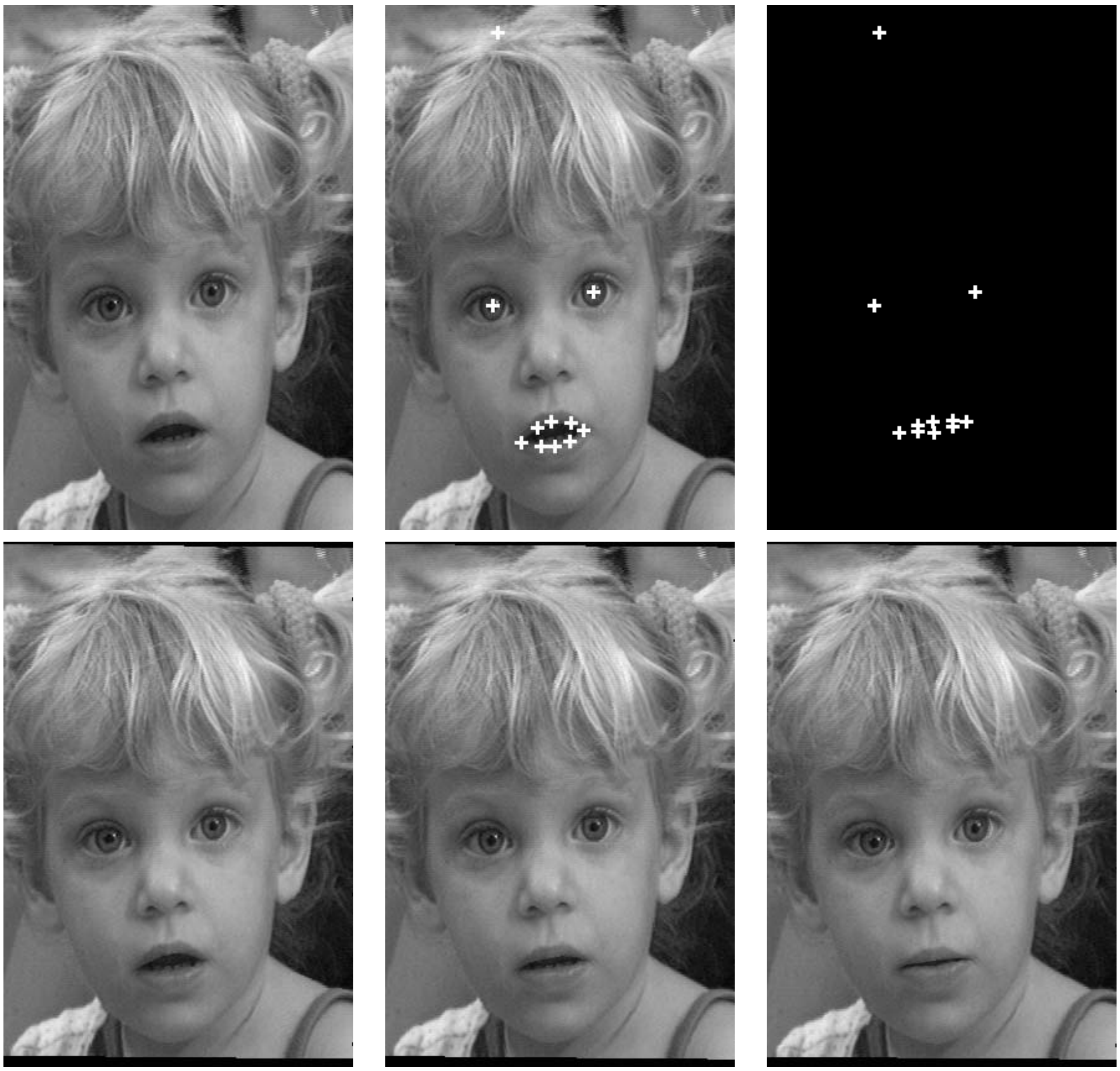

Figure 6: Animation using **only one** snapshot. **Top**: Original image, source location of the anchor points marked on the image and destination of the anchor points. **Bottom** (left to right): Transformations using Gaussian radials with $\sigma = 0.25$ and the tradeoff parameter $\lambda = 10^{-3}, 10^{-4}, 0$ respectively.

Figure 7: An unrealistic image caused by using thin plate radials instead of the Gaussian radials used in the previous figure.

for each function evaluation. Overall the time needed for computing the value of a basis function at a given point is of the order of the timw needed for an indirection addressing mode.

In order to evaluate the mapping at a given point, one has to compute the value of the radial function at N, number of anchors, points or less. When using a local-influence basis function, each image point is influenced only by its neighboring anchor points, and in particular large areas in the image are not influenced by any anchors. For each influencing anchor an indirection (for evaluating the radial) plus a multiplication (by the proper coefficient) is needed. Thus, in any practical application the overall number of operations needed is a small number of multiplications and indirections times the image size.

When using thin-plate spline, an alternative to using look-up tables is to use certain linear combinations of the original basis functions that decay polynomially; i.e., another basis (not necessarily radial) can be constructed using functions f, satisfying $f(t) = O(t^{-k})$ [8], We, however, were satisfied with look-up tables, and thus did not further persue this approach.

5 Discussion

5.1 Comparison with other Works

In the last several years a consirable amount of research has been directed towards image warping in general and animation of facial expressions in particular. From our point of view models and procedures intended at facial animation may be classified into two families: The first – *model dependent*, generates facial expressions by first constructing a mathematical model of the physical face, and then defining the dynamics which govern the non-rigid motion of the object, e.g. [22, 28]. These techniques may exhibit impressive results, but suffer from the fundamental drawback that they are object dependent, i.e. a different model is needed for different non-rigid objects. The second family of techniques – *model independent* simulate deformations without using any information on the object being deformed [27, 17, 13]. Recently these two approaches have been combined [23], in the sense that an association between mathematical parameters defining the transformations and real-life facial expressions was established, giving rise to an expression editor. Ideally, model dependant warps mimic the real world realistically, however at the present they are much simpler than the real world. The lack of knowledge of the transformation of complex objects renders the construction of accurate model-dependant warps unsatisfactory. The present work falls within the category of model independent transformations.

Of crucial importance in our model is the small number of points needed in order to define the warp, their general position, and the generalization of affine mapping which is an integral component of the transformation.

Another model that defines a transformation by the position of anchor points is the *free-form deformation* model of Sederberg and Parry [27]. In this model the anchors (control points) must lie on a regular grid, thus imposing at least 4 control points in the planar case, but typically many more. Moreover, the position of the points may not coincide with the position of physical features that are to be manipulated.

Another algorithm that has cumulated in an impressive Michael Jackson video is the *feature based image metamorphosis* algorithm [1], where the position of each point is the weighted average of affine transformations determined by corresponding line segments in the source and target images. Since each line segment corresponds to two points, this mapping is again driven by the position of at least 4 anchor points, and in any case an even number of points are needed. Thus a general affine transformation cannot be realized by this method. We feel that the use of pairs of points (line segments) is not as natural as using single points to specify the transformation. In addition, segments cannot intersect, thus not every configuration of anchor points is possible. Finally, the weight functions governing the effect of each segment on the mapping are not local, and each pair of anchors has a global effect. This last point may be modified by using locally

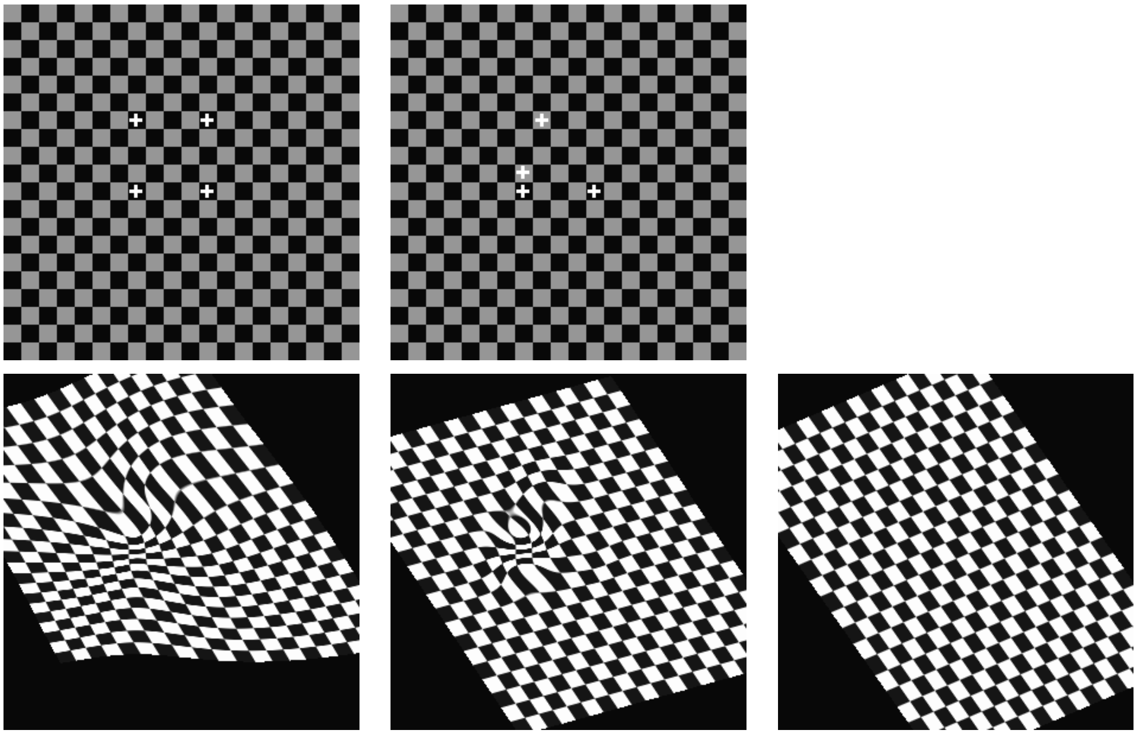

Figure 8: Generalizing affine mappings in different ways. **Top**: Position of source and target anchor points. **Bottom** (left to right): thin-plate warp, Gaussian warp, affine least-square warp ($\lambda = \infty$). In all cases the mapping can be well approximated by an affine mapping far away from the anchors. In the thin-plate case this affine map is different at different regions, unlike the Gaussian case in which the same affine component appearing in the definition of the mapping dominates the transformation in all areas away from the anchors.

supported weight functions.

A model-free warping algorithm driven by the position of source and target anchor points is the *nonlinear mappings for modeling of geometric details* of van Overveld [29]. Like our model, the warp is affine whenever possible. However, when more than three anchors are used, it is rare that affine warping is possible, and a definition of a generalization of affine warping is needed. Unlike the Overveld model, in our model this is realized by the fact that an explicit affine component is always present. The use of the Gaussian basis function enables the affine component to dominate the mapping away from the anchors. While using the thin-plate radial, a minimal deviation from the affine family of mappings, in the sense of bending energy, is achieved. Finally, the rich mathematical structure of radial basis function theory enables additional generalizations such as the incorperation of non-exact interpolation, demonstrated above. The significance of these generalizations is demonstrated in figure 8.

5.2 Image warping and face recognition

Many applications that involve face images, and in particular face recognition tasks, require normalization of faces. Invariance to lighting and viewing conditions is traditionally handled by classical computer vision methods, but the variability caused by facial expression is not. Thus, we have tried to construct a facial expression transformation without an explicit 3D face model, namely to construct a $\mathbb{R}^2 \to \mathbb{R}^2$ mapping of (facial) images that can handle a large number of facial expressions, using a minimal number of constraints. These constraints should rely on the position of few anchor points that can be detected automatically.

There are various alternatives for the construction of such a mapping. Functions of a complex variable have been suggested in such a setting [13] since $\mathbb{R}^2$ is naturally incorporated into their structure. Analyticity is a primary attribute of such functions, meaning that they are conformal transformations. Therefore these functions cannot reproduce a general affine mapping, a requirement which is paramount in our case.

Another possibility is constructing a pair of $\mathbb{R}^2 \to \mathbb{R}$ functions where each of these functions is constructed by a tensor product of univariate functions. Such an approach involves tessellations of the plane, and requires a relatively large number of points to be specified [3, 5, 12]. This drawback applies to other tessellation dependent techniques, such as an adaptive meshing technique [30] which uses a facial muscle control model defined on a mesh.

Turning to the family of mappings defined in the present work, 2D radial basis function transformations overcome the drawbacks previously mentioned. They are specified by three or more points, generalize affine mappings, and satisfy the global requirements while preserving the local effect of each anchor point on its neighborhood. There is a rivalry between the locality requirement and global constraints. The choice of the specific radial basis function influences the tradeoff factor between these notions. The use of the thin-plate basis emphasizes the global nature of the mapping, while the Gaussian basis enables the stressing of locality, as we have demonstrated. These two families of basis functions are by no means the only ones possible. We have held experiments with other base functions such as t^α , $1 < \alpha < 2$, with comparable success.

5.3 General remarks

A major motivation for this work was the successful use of affine transformations in face normalization [10]. One feature of affine mappings is their group structure: The family of affine transformations is closed under composition and inversion. This structure is attractive in interactive systems, since the position of the anchor points can be successively tuned. Thus, the final outcome is memoryless. It has also a useful role in overcoming aliasing problems as discussed in the previous section. We have shown that radial basis mappings generalize the affine transformation in various respects, but the group structure is destroyed. Nonetheless, successive applications of radial basis mappings is possible, and the final outcome is comparable to the one shot mapping determined by the total displacement of the anchor points in case the displacements are small.

We have emphasized the application of this technique to face images, although it is valid for other images of elastic objects, since the aforementioned constraints hold for such cases as well.

The method we propose in this work can be employed to map face images of various expressions to standard facial templates. Thus, an appropriate selection of the number of anchor points, their approximate location, and the choice of the basis function can facilitate a 2D model of facial expressions. Further research may give explicit interpretation in facial expression terms (such as smiles, frowns, etc.) to the exact parameters chosen.

Acknowledgments

We thank Daniel Cohen and David Levin for valuable comments and discussions. The work was supported by grant No. BSF8900078 from the US-Israeli Binational Science Foundation.

References

[1] T. Beier and S. Neely. Feature-based image metamorphosis. *Computer Graphics*, 26(2):35–42, 1992.

[2] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:567–585, 1989.

[3] C. De Boor. On calculating with B-splines. *Journal of Approximation Theory*, 6:50–62, 1972.

[4] H. H. Bülthoff and S. Edelman. Psychophysical support for a 2D view interpolation theory of object recognition. *Proceedings of the National Academy of Sciences*, 89:60–64, 1992.

[5] W. Dahmen and C. A. Micchelli. On linear independence of B-splines I: Triangulations of simploids. *SIAM journal of Numerical Analysis*, 19:993–1012, 1982.

[6] J. Duchon. Splines minimizing rotation-invariant semi-norms in Sobolev spaces. In C. K. Chui, L. L. Schumaker, and J. D. Ward, editors, *Multivariate Approximation Theory*, pages 85–100. Birkhauser, Basel, 1979.

[7] N. Dyn. Interpolation and approximation by radial and related functions. In C. K. Chui, L. L. Schumaker, and J. D. Ward, editors, *Approximation Theory VI*, volume 1, pages 211–234. Academic Press, 1989.

[8] N. Dyn, D. Levin, and S. Rippa. Numerical procedures for global surface fitting of scattered data by radial functions. *SIAM Journal of Scientific and Statistical Computing*, 7:639–659, 1986.

[9] N. Dyn and G. Wahba. On the estimation of functions of several variables from aggregated data. *SIAM Journal Of Mathematical Analysis*, 13(1):134–152, 1982.

[10] S. Edelman, D. Reisfeld, and Y. Yeshurun. Learning to recognize faces from examples. In *Proceedings of the 2nd European Conference on Computer Vision*, pages 787–791, Santa Margherita Ligure, Italy, May 1992.

[11] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer graphics: principles and practice*. Addison-Wesley, 1990.

[12] P. Fong and H. P. Seidel. Control points for multivariate B-spline surfaces over arbitrary triangulations. *Computer Graphics Forum*, 10:309–317, 1991.

[13] C. Frederick and E. L. Schwartz. Conformal image warping. *IEEE Computer Graphics and Applications*, March, 1990:54–61, 1990.

[14] F. Galton. Composite portraits, made by combining those of many different persons, into a single, resultant figure. *Journal of the Anthropological Institute*, 8:132–144, 1879.

[15] F. Galton. Personal identification and description – ii. *Nature*, pages 201–203, 22 Jun 1899.

[16] P. S. Heckbert. Survey of texture mapping. *IEEE Computer Graphics and applications*, 6(11):56–67, November 1986.

[17] Z. C. Li, C. Y. Suen, T. D. Bui, and Q. L. Gu. Harmonic models of shape transformations in digital images and patterns. *CVGIP: Graphical Models and Image Processing*, 54(3):198–209, May 1992.

[18] W. R. Madych and S. A. Nelson. Multivariate interpolation: a variational theory.

[19] N. Magnenat-Thalmann, P. Primeau, and D. Thalmann. Abstract muscle action procedures for face animation. *The Visual Computer*, 3:290–297, 1988.

[20] G. A. Micchelli. Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22, 1986.

[21] M. Oka, K. Tsutsui, A. Ohba, Y. Kurauchi, and T. Tago. Real-time manipulation of texture mapped surfaces. *Computer Graphics*, 21(4):181–188, 1987.

[22] F. I. Parke. Parameterized models for facial animation. *IEEE Computer Graphics and Applications*, pages 61–68, November 1982.

[23] P.Kalra, A.Magnili, N. Magnenat Thalmann, and D. Thalmann. Simulation of facial muscle actions based on rational free form deformations. *CGF*, 11(3):C–59 – C–69, 1992.

[24] S. Platt and N. I. Badler. Animating facial expression. *Computer Graphics*, 15(3):245–252, 1981.

[25] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Detection of interest points using symmetry. In *Third International Conference on Computer Vision*, pages 62–65, Osaka, Japan, December 1990.

[26] D. Reisfeld and Y. Yeshurun. Robust detection of facial features by generalized symmetry. In *Proceedings of the 11th International Conference on Pattern Recognition*, The Hague, Netherlands, September 1992.

[27] T. W. Sederberg and S. R. Parry. Free-form deformation of solid geometric models. In *SIGGRAPH '86*, pages 151–160, 1986.

[28] D. Terzopolous and K. Waters. Physically based facial modeling, analysis, and animation. *Journal of Visualization and Animation*, 1(2):73–80, 1990.

[29] C. W. A. M. van Overveld. Beyond bump maps: nonlinear mappings for the modeling of geometric details in computer graphics. *Computer Aided Design*, 24(4):201–209, April 1992.

[30] K. Waters and D. Terzopoulos. Modeling and animating faces using scanned data. *The Journal Of Visualization and Computer Animation*, 2(4):123–128, 1991.

[31] L. Williams. Performance-driven facial animation. *Computer Graphics*, 24(4):235–242, 1990.

[32] J. F. S. Yau and A. D. Duffy. A texture mapping approach to 3-D facial image synthesis. *Computer Graphics Forum*, 17:129–143, 1988.