

Enhancement by Image-Dependent Warping

Nur Arad and Craig Gotsman

Abstract—All image warping algorithms to date are image-independent, namely, relate only to the geometry of the image plane, ignoring the content of the image. We show that taking the image content into account yields elaborate warping schemes which may be used to enhance, sharpen and scale images. Sharpening the image is achieved by “squashing” the pixels in edge areas, and “stretching” the pixels in flat areas. Since image pixels are only moved, not modified, some drawbacks of classical linear filtering methods are avoided. We also lay the mathematical foundation for the use of an image-dependent warping scheme in traditional warping applications, such as distortion minimization.

Index Terms—Enhancement, image warping, variational surface fitting.

I. INTRODUCTION

IMAGE warping is an active branch of image processing and computer graphics, in which a given image is geometrically distorted to conform to given specifications. Image warping has diverse applications, mainly for registration purposes in cartography, remote sensing and tomography [1], and for creating visual special-effects in the entertainment industry. It is closely related to the popular “morphing” technique (see, e.g., [2], [3]). In most interactive warping systems, the user specifies the warp in some very general way, e.g., by moving grid lines [4], [5] or by specifying a point-to-point correspondence [6]–[8], and then the computerized system automatically interpolates this geometric specification S to produce a mapping $W_S: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ of the plane to itself. The mapping is then applied to the input image, i.e., if $I(x, y)$ is the input image, the output image $O(x, y)$ is defined by $O(W_S(x, y)) = I(x, y)$ (forward mapping) or $O(x, y) = I(W_S^{-1}(x, y))$, where W_S^{-1} is the functional inverse of W_S (backward mapping) [9]. Contemporary warping algorithms produce a W_S which does not depend on I , namely, once the specification S has been provided, the mapping W_S may be computed without knowledge of the image content. So, even though the geometric specification S is based on image features, the complete W_S is not.

The key question in the design of an image warp is how to produce W_S from the specification S . In general, this problem is ill-posed, since there are usually many possible warps satisfying the constraints imposed by S . A common approach

to obtaining W_S is by *regularization*, namely, solving for the warp minimizing some “roughness” criteria. An example [10]–[12] is the *thin-plate spline* W_S which is the unique mapping satisfying S and which is also the solution u to

$$\arg \min_{u(x,y)} \iint_{\mathbb{R}^2} (u_{xx}^2 + 2u_{xy}^2 + u_{yy}^2) dx dy \quad (1)$$

where u_{xx} denotes $\partial^2 u / \partial x^2$, and likewise for u_{xy} and u_{yy} . Computing the warp using this approach involves the solution of a set of linear equations whose size is equal to the number of constraints in S . Another approach to the generation of W_S is to construct a warp giving rise to fast and straightforward implementation while giving geometrically intuitive results [7], [13].

The idea behind minimizing a functional such as (1) is to produce a warp which minimizes “distortion” to the image. As such, the fact that W_S is independent of I can sometimes lead to suboptimal results. Indeed, the mapping function guarantees the minimization of the functional over the domain. However, the mapping is only an intermediate stage in the entire procedure, and after its computation it is ‘applied’ to the image I . Thus its roughness is apparent only in ‘busy’ areas of the image. Constructing mappings which have a higher value of the functional, but in which the larger values of the integrand are concentrated in smooth areas of the image may give rise to results with a smoother appearance. Clearly this may be done only if image content is taken into account. This paper explores this idea.

An application of the image-dependent warping technique presented here is image scaling combined with enhancement. It provides a methodology quite different from existing ones. Image enhancement is a general term for a variety of techniques which improve the visual quality of an image. The standard approach to image enhancement is to modify the *intensity* of pixels based on intensities of their neighbors, by applying linear filtering (which may in fact be an implementation of a nonlinear method [14]) or by invoking adaptive histogram-based methods [15]. We propose not to operate on pixel intensities, but rather modify their position in the image plane. By local “stretching” and “squashing,” we are able to achieve various effects. Fig. 1(h), compared to the original signal—Fig. 1(a), illustrates how a one-dimensional edge may be “sharpened” by a transformation of the coordinate system (a one-dimensional (1-D) warp) which “squashes” the signal in the vicinity of the edge, and “stretches” it in other areas. In comparison, the popular “unsharp masking” technique, which is a linear filtering technique, modifies the signal f by the simple rule $f \mapsto \alpha f + \alpha f$ for some small α , and produces a “ringing” Gibbs effect. Our approach is somewhat reminiscent

Manuscript received June 11, 1997; revised October 21, 1998. This work was performed while the authors were with Hewlett-Packard Israel Science Center, Technion City, Haifa 32000, Israel. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Henri Maitre.

N. Arad is with Orbotech Ltd., Yavne 81102, Israel.

C. Gotsman is with the Computer Science Department, Technion—Israel Institute of Technology, Haifa 3200, Israel (e-mail: gotsman@virtue3d.com).

Publisher Item Identifier S 1057-7149(99)06104-7.

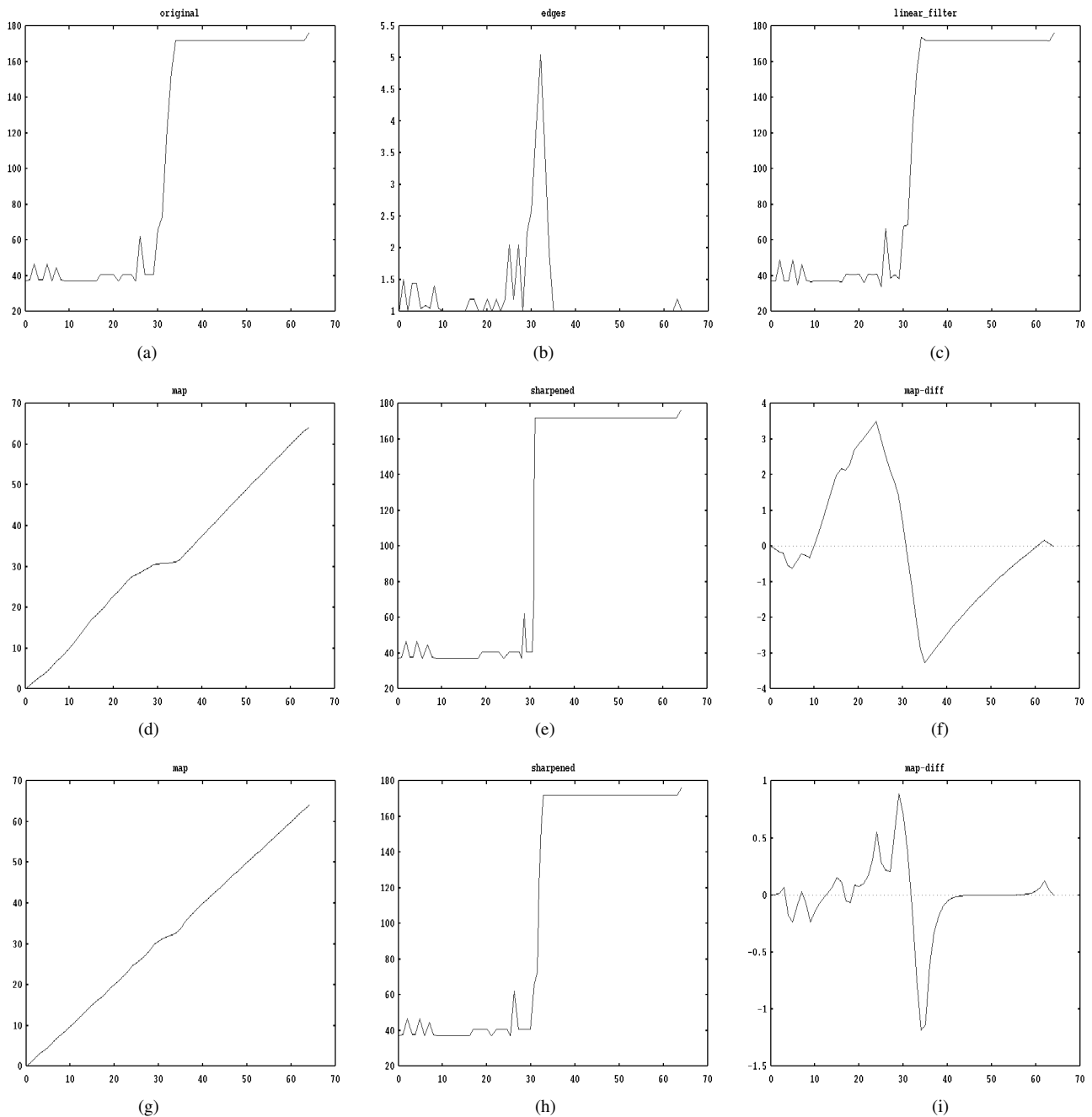


Fig. 1. Example of 1-D edge enhancement (scaling factor $m = 1$). (a) Original signal. (b) The image-dependent component $\mathcal{I} = 1 + \lambda|I'|$. (c) Result of applying the linear “unsharp masking” filter. Notice the Gibbs effect at the edges. (d) Mapping function with $\lambda = 0.1$ and clamping parameter $\mu = 0$. (e) Result of applying mapping (d) to the original signal. (f) Graph showing movement of pixels in (d) relative to uniform scaling. As can be seen, maximal movement is almost four units. (g) Mapping function with $\lambda = 0.1$ and clamping parameter $\mu = 0.2$. (h) Result of applying mapping (g) to the original signal. (i) Graph showing movement of pixels in (g) relative to uniform scaling. Maximal movement is reduced to one unit, and prominent edges remain fixed.

of the anisotropic diffusion methods [14] in the sense that smoothing is performed inside regions depicting an object while the intra-region edges are enhanced.

This paper is organized as follows: Section II describes in detail the proposed method. We start out with an analysis of a univariate reduction of our problem and then show the generalization to the two-dimensional (2-D) setting. Section III gives some examples of the performance of the method and discusses the run-time in a simulation environment. Section IV discusses mathematical issues involved in the implementation. Finally, Section V relates the present work with other image-

enhancement methods on the one hand and image-warping methods and applications on the other.

II. APPLICATION 1: IMAGE UPSCALING

Regularization-driven warping techniques have been used to date for image registration and computer graphics special effects. We now consider using the same approach for image scaling. The technique we propose may be better understood by first examining a 1-D example. The reduction to one

dimension will also serve us in formulating the bivariate problem.

A. Signal Scaling (1-D Case)

Suppose we want to scale a 1-D signal by a factor of m in such a way as to preserve the sharpness of the edges. A simple (uniform) scaling procedure would be written as $O(mx) = I(x)$, where m is the scaling factor. Translated to image warping terminology, the warping function is $W(x) = mx$. Applying this warp will, in general, blur ‘edges’ in the signal. Preserving the sharpness of the edges entails nonuniform scaling with edge areas “squashed,” while other areas are stretched. This may be achieved by constructing a 1-D warp $W_{S,I}$, which is the solution of the following variational problem:

$$\min_{u(x)} \int_0^1 (u'(x))^2 [\Phi(|I'(x)|)] \quad S = \{u(0) = 0, u(1) = m\}. \tag{2}$$

Φ should be a continuous, positive and increasing function. We have chosen $\Phi(|I'(x)|) = 1 + \lambda|I'(x)|$, where λ is a user-defined positive parameter and the original signal, $I(x)$, is defined on $[0, 1]$. The larger the value of λ the more pronounced the sharpening effect near the edges will become. The boundary conditions force a global scaling factor of m , while the variational constraint induces small values on the derivative of the warping function where $|I'(x)|$ is large and vice-versa. The solution reduces to uniform scaling in the case $I' \equiv \text{const}$.

Integrating by parts, one arrives at an equivalent differential formulation of (2):

$$[u'(x)(1 + \lambda|I'(x)|)]' = 0, \quad u(0) = 0, \quad u(1) = m \tag{3}$$

whose solution may be given in closed form as

$$u_{S,I}(x) = m \left(\int_0^1 \frac{dt}{\mathcal{I}(t)} \right)^{-1} \int_0^x \frac{dt}{\mathcal{I}(t)} \tag{4}$$

with $\mathcal{I}(x) = 1 + \lambda|I'(x)|$.

Despite the simplicity of the solution (4) it suffers an inherent drawback: Due to the global nature of the mapping function, the movement of samples in the signal is not bounded, yielding unwanted effects [see Fig. 1(d)–(f)]. To overcome these effects, a *clamping parameter* $\mu > 0$ is incorporated into the variational formulation of the problem:

$$\min_{u(x)} \int_0^1 ((u'(x))^2 + \mu(u(x) - mx)^2) \mathcal{I}(x) dx, \tag{5}$$

$$S = \{u(0) = 0, \quad u(1) = m\}$$

The sharpening effect holds as before, but now the clamping parameter forces the position of edge points to be virtually scaled by the exact factor of m . An example of the importance of the clamping parameter may be seen in Fig. 1(g)–(i).

Losing the compactness of the closed-form solution (4), Problem (5) may still be converted to a second-order differential equation

$$[u'(x)\mathcal{I}(x)]' = \mu(u(x) - mx)\mathcal{I}(x), \quad u(0) = 0, \quad u(1) = m \tag{6}$$

whose solution is obtained, in any practical case, by numerical methods (see Section IV).

B. Image Scaling

When attempting to scale a 2-D signal, the same principles as in the 1-D case apply. The only additional notion needed is that of *separability*: Most contemporary warping procedures are *separable* image warps, in the following sense: The warping function is represented as $W_S = (W_S^X, W_S^Y)$ —a pair of bivariate real functions, which are in the general case, produced independently. When one, for example, invokes the thin-plate regularization paradigm in the design of a warp, one decomposes the set of interpolation constraints S into x -constraints and y -constraints, and constructs the mapping functions W_S^X and W_S^Y , each minimizing the warping functional (1), and also satisfying their respective set of constraints. We will also be using the separable framework in our construction.

An image warp $W_{S,I}(x, y)$ that squeezes the edges while stretching other areas may be obtained as the solution to the following optimization problem:

$$\min_{u(x,y)} \iint_{[0,1]^2} [u_x^2 + u_y^2 + \mu(u(x, y) - m(x, y))^2] \cdot \mathcal{I}(x, y) dx dy \tag{7}$$

where for the x -component we substitute $m(x, y) = mx, \mathcal{I}(x, y) = 1 + \lambda E_x(x, y)$, and for the y -component we have $m(x, y) = my, \mathcal{I}(x, y) = 1 + \lambda E_y(x, y)$. $E_x(x, y)$ and $E_y(x, y)$ are the x - and y -components of a particular implementation [16] of the Canny edge-detector [17], both taken to be normalized such that $0 \leq E_x(x, y), E_y(x, y) \leq 1$. The Canny edge-detector is by no means instrumental, and may be replaced but any reasonable continuous-valued detector. Li *et al.* [18] use a similar functional in an image-independent ($\mathcal{I} \equiv 1, \mu = 0$) patch-warping scenario. At a first glance such a formulation seems to lead to a solution which is not rotation invariant, thus potentially leading to grid artifacts. However assuming the image dependent-component is rotation invariant (more precisely, commutes with rotation—a reasonable constraint) the solution of (7) possesses the same property, since it minimizes a functional which is rotation invariant. This observation assumes that the boundary conditions may be formulated in a rotation-invariant manner, which is quite difficult in practice. However, the effect of the boundary conditions decays quickly away from the boundary of the image, thus the whole procedure is in practice indeed rotation-invariant. The same principle (minimization of a rotation-invariant functional) allows the use of separable thin-plate warping in classical warping applications [10].

Although of limited effect far away from the boundaries, special care must still be taken when constructing the boundary conditions. Clearly we must have $\{W_{S,I}(x, 0) | 0 \leq x \leq 1\} = \{(mx, 0) | 0 \leq x \leq 1\}$, with similar equalities for the other three boundaries. Note that these are point-set equalities. However, in order to arrive at a well-posed and tractable problem, we must specify the location of every point on the domain boundary. The trivial solution of linear scaling,

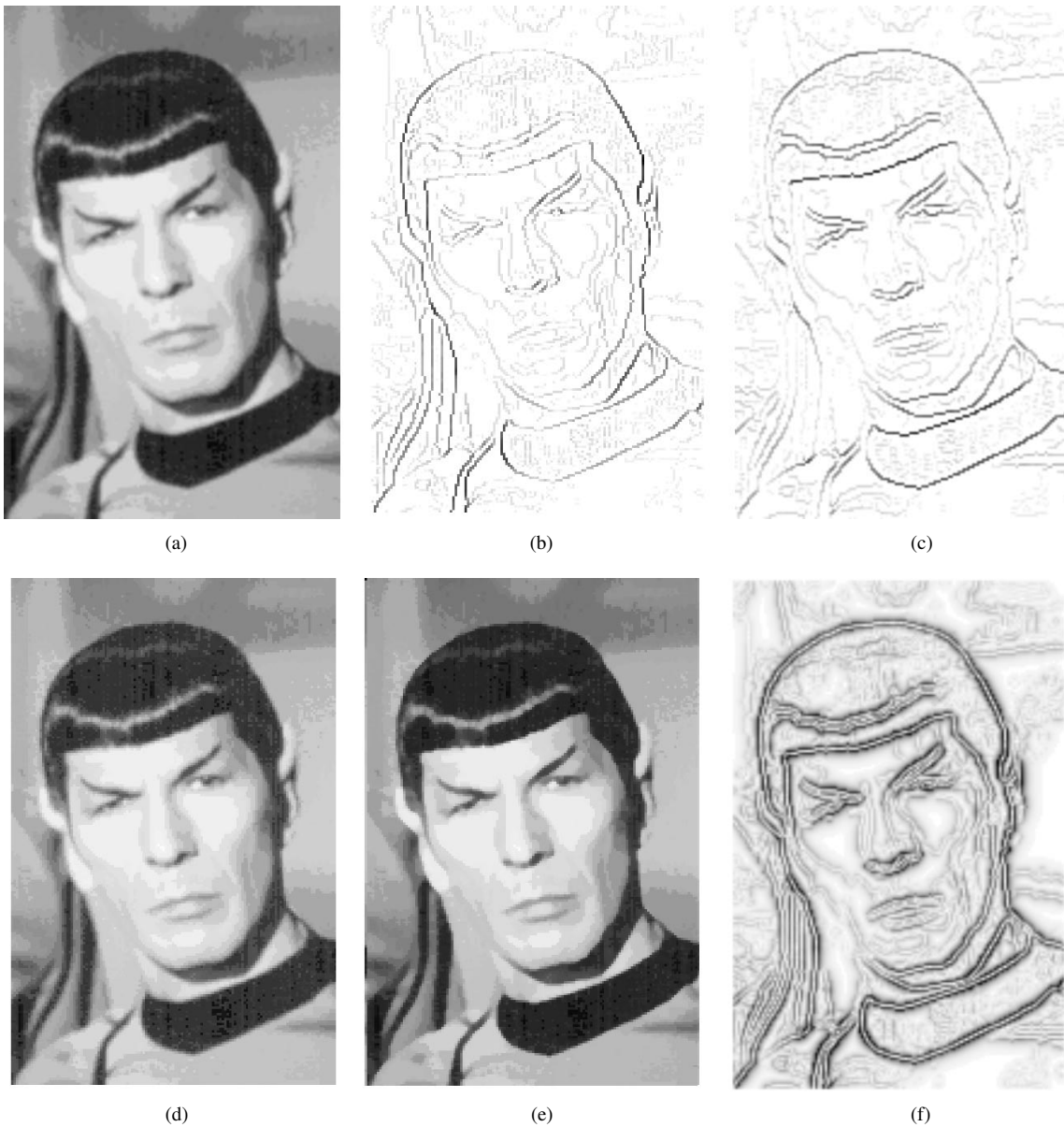


Fig. 2. Basic application and intermediate steps in application of the proposed algorithm. (a) Original image. (b) Image-dependent weighting factor used for the horizontal warp (white = 1, black = maximal value). (c) Image-dependent weighting factor used for the vertical warp. (d) Result of applying unsharp masking. Notice the clamping combined with the Gibbs effects at the hairline and collar areas. Result is of higher contrast and noisier, but edges are not sharper. This result was obtained by use of the popular xv imaging package. The original image was highly quantized. (e) Result of proposed enhancement with $\lambda = 2$ and $\mu = 0.1$. (f) Image shows movement of pixel positions relative to their origin (white = no movement). Highest value is 0.7, and is obtained on both sides of the edges.

$W_{S,I}(x,0) = (mx,0)$, was found to give unsatisfactory results, since it allows no ‘edge-squeezing’ on the boundary of the image. On the other hand, specifying the boundary conditions as the reduction of the problem to the univariate case (Section II-A) creates too strong boundary effects. As a compromise, we may take the boundary value $W_{S,I}(x,0)$ to be the solution of the univariate problem (5) with the expression $1 + \lambda|E_x(x,0)|$ acting as the image-dependent component. The other boundary values are calculated in a similar manner.

Using standard techniques from the calculus of variations [19], one may again show that minimizing the integral in (7) is equivalent to solving the differential equation

$$(u_x \mathcal{I})_x + (u_y \mathcal{I})_y = \mu(u(x,y) - m(x,y)) \mathcal{I} \quad (8)$$

\mathcal{I} again denoting the image-dependent component, with the same boundary conditions. This is an elliptic second-order partial differential equation, and as such admits to highly efficient numerical solutions [20].

C. Image Enhancement

A special case of image magnification is when $m = 1$. At first glance it would seem that the image should not be affected. However, just as the edges are preserved for the case $m > 1$, a similar edge-enhancing effect may be achieved for $m = 1$, which is useful in its own right. Fig. 2 shows the results of our image enhancement procedure compared to the standard unsharp masking technique. Note that in the current framework, enhancement ($m = 1$) and scaling ($m > 1$)

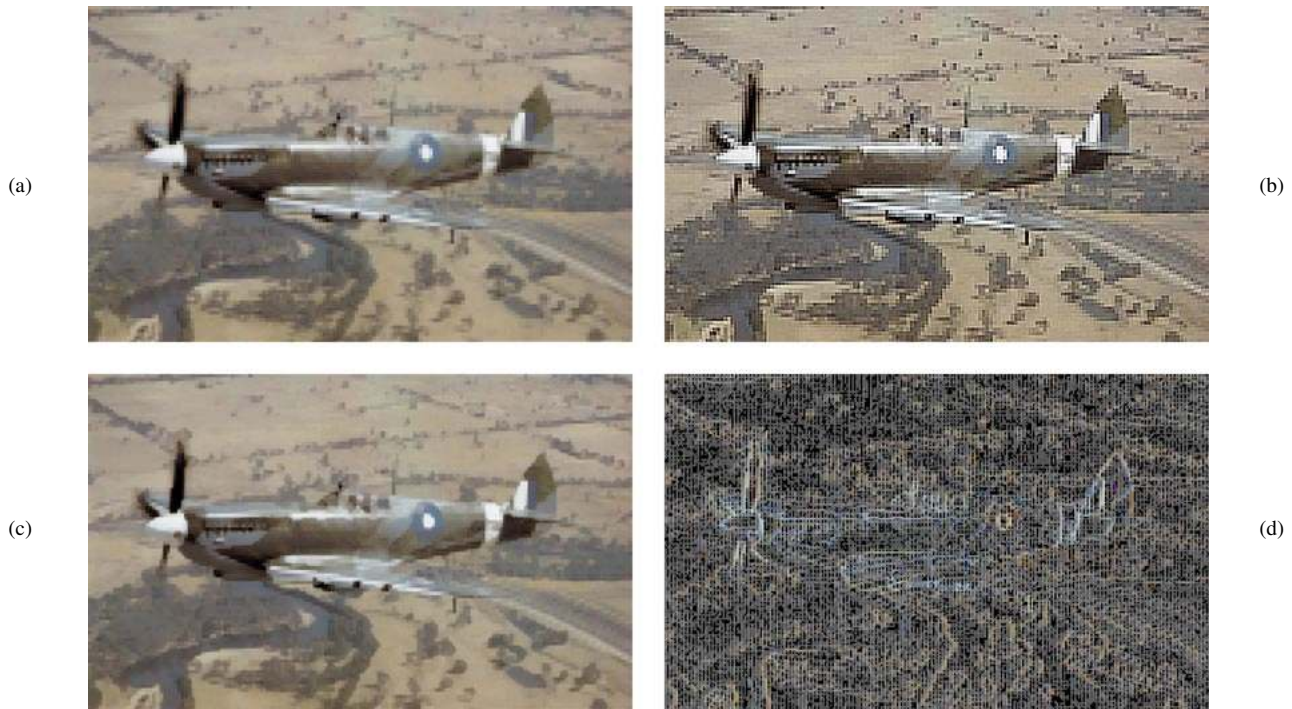


Fig. 3. (a) Original image of famous WWII fighter. (b) Application of unsharp masking to the image. Again note that contrast is improved, but not edge resolution. (c) Result of applying the enhancement warp with $\lambda = 2$. Note the sharp propeller and tail. (d) RGB differences between (a) and (c) gamma-corrected with a value of $\gamma = 4$.

are treated in the same manner: Assuming a scale-invariant edge detector is used, and due to the linearity of the problem [minimizing (7) is equivalent to solving a sparse linear system] scaling may be performed *after* enhancement with the scaling factor $m = 1$, resulting in a considerable reduction in run-time. This is equivalent to solving the linear system with a multiscale paradigm.

III. EXAMPLES

Fig. 2 shows the intermediate stages and the results of applying the enhancement by image warping procedure to the image of a well-known personality. Notice that as opposed to linear image processing methods, no pixel value overflow occurs, hence no cropping is needed.

Fig. 3 shows the same procedure again together with the result of unsharp masking. Fig. 4 shows the application of the proposed method for scaling, and again is compared with a traditional linear image processing approach. Fig. 5 is yet another visualization of the pixel-position movement. Note that this figure should not be judged by its visual merits, since it is intended only as a visualization of a particular step in the whole procedure.

In general one may say that the proposed method smooths out homogeneous regions while sharpening the border between distinct regions. The smoothing of the homogeneous regions is not intrinsic to the method, and is actually an artifact of the bilinear resampling step. A more advanced resampling scheme such as parametric cubic convolution [21] may well decrease the smoothing effect exhibited in these examples.

Table I summarizes run times for the examples shown in this report. The algorithm was run on an HP 715/75 workstation

TABLE I
RUN-TIME OF THE ALGORITHM ON THE EXAMPLES SHOWN

image	type	size	scale	run-time
Spock	8-bit gray	139 × 212	×1	1.5 sec.
Spitfire	24-bit color	307 × 190	×1	4.25 sec.
Cat	8-bit gray	128 × 122	×2	1.31 sec.
Pfeiffer	24-bit color	262 × 215	×4	13.17 sec.

using unoptimized C++ code. The images were loaded from storage in binary *pgm/ppm* format, and results were not saved.

One may note that processing time is divided between computing the mapping between source and target pixels, which depends only on image size, and the resampling stage which depends on the number of channels used (grayscale versus 24-bit color) and the scaling factor used. Thus the Pfeiffer image takes a tenfold increase in run-time relative to the Spock image mainly due to number of channels and scaling used. Considerable acceleration may be obtained during the resampling stage by use of look-up tables and single-instruction multiple-data (SIMD) technology. These run times are considerably larger than the application of a simple linear filter, but the differences become less pronounced when a high scaling factor is used, since then the resampling step consumes the majority of the processing time.

IV. IMPLEMENTATION DETAILS

A. Forward versus Backward Implementation

As stated earlier, supposing the mapping function $W: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ has been computed, there are two methods of implementing it in order to generate the

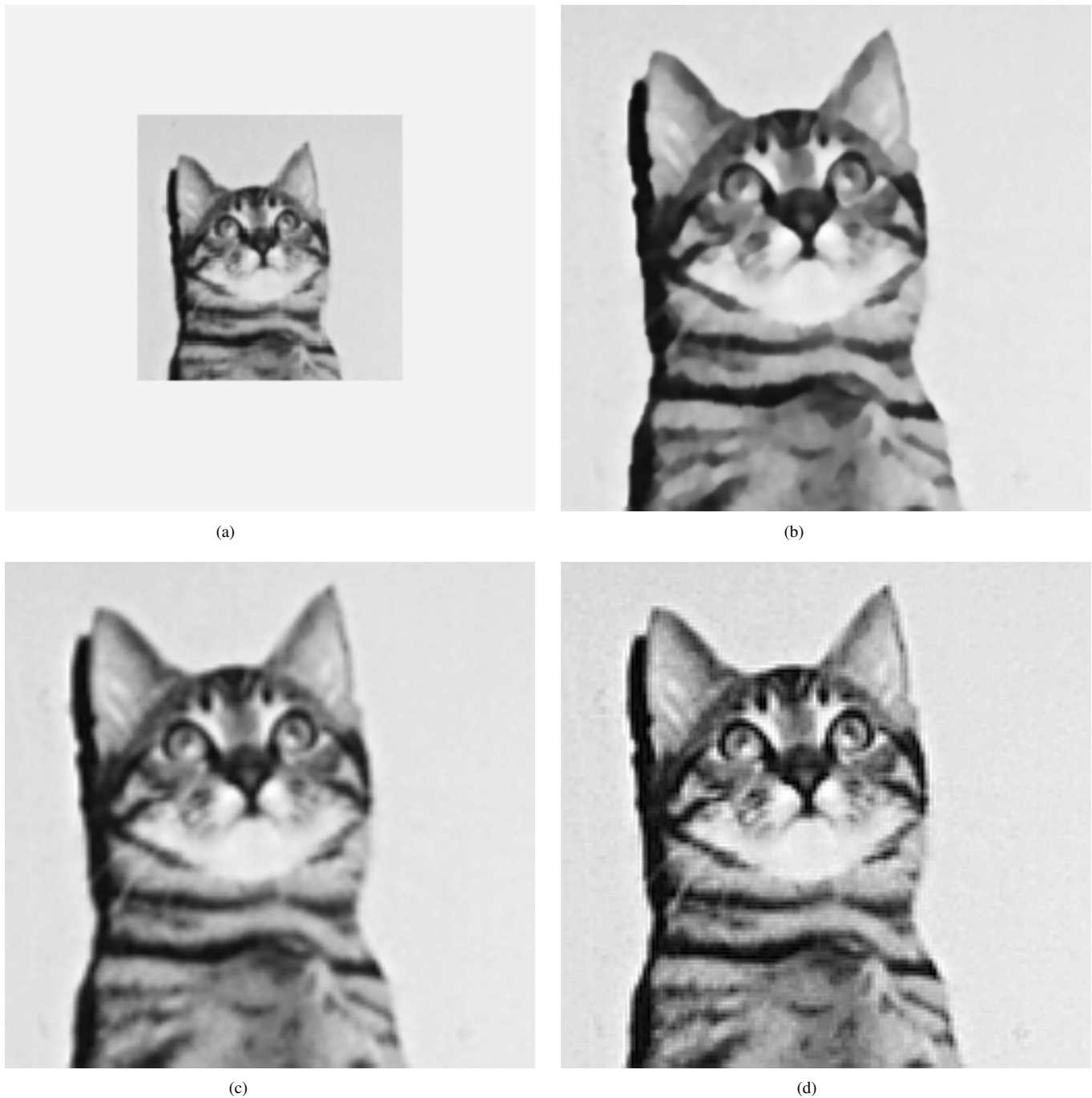


Fig. 4. Scaling by factor $m = 2$. (a) Original image of tabby cat. (b) Scaling by method proposed. (c) Scaling by bilinear interpolation. (d) Application of unsharp masking on the result of the bilinear interpolation output. Using unsharp masking before scaling leads to similar results. This example shows that the proposed method is biased toward piecewise continuous solutions and in this sense leads to results similar to the anisotropic diffusion process of Perona and Malik [14].

output image $O(x, y)$; forward— $O(W_{S,I}(x, y)) = I(x, y)$ and backward— $O(x, y) = I(W_{S,I}^{-1}(x, y))$. Backward implementation is to be preferred provided the inverse mapping $W_{S,I}^{-1}$ is known and the input image is available in random access. Unfortunately, in our case $W_{S,I}^{-1}$ is not available in closed form, a problem present in other warping techniques [7]. In many warping situations this may be solved by computing the mapping induced by inverting the specification S . However, in our case the specification S is trivial and (up to the scaling factor) equals its inverse. To

solve this problem, (7) is modified as follows:

$$\min_{u(x,y)} \iint_{[0,1]^2} [(u_x^2 + u_y^2)/\mathcal{I}(x, y) + \mu(u(x, y) - m^{-1}(x, y))^2 \cdot \mathcal{I}(x, y)] dx dy \quad (9)$$

with $\mathcal{I}(x, y)$ as in (7) (for the x -component). The solution is taken as an approximation to the inverse of the desired mapping. Notice that *inverse* stretching and squashing (needed due to the backward implementation) is obtained by the

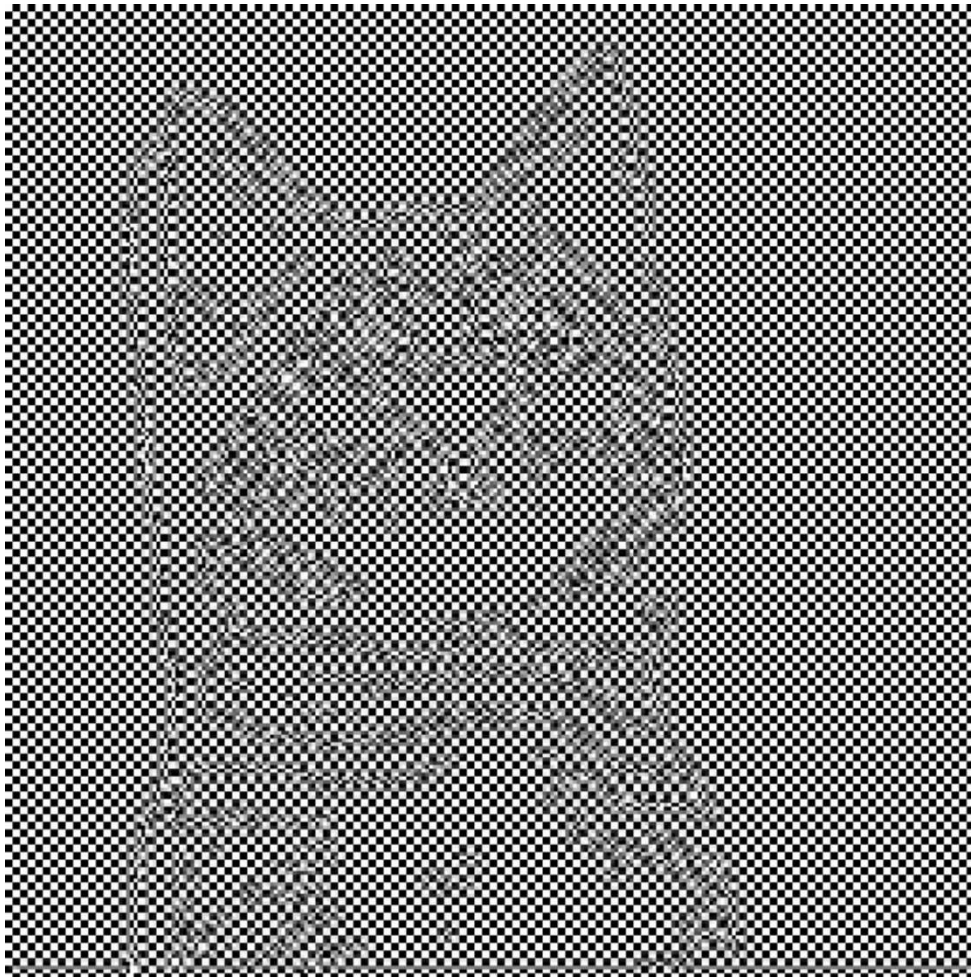


Fig. 5. Application of the image-dependent warp produced for Fig. 4 on a checkerboard image, in which the movement of the pixels is clearly evident.

reciprocal of the image-dependent component— $1/\mathcal{I}$, while the clamping is achieved with the direct component— \mathcal{I} . One must also bear in mind that when implementing backward, the image-dependent component should be computed from the (still unknown) target image. Fortunately, in the present setting the image-dependent components of the source and target images are practically identical (they are both edge-detector outputs of the image, original or enhanced, and as such, one may be considered as a good approximation to the other). This point may very well prohibit reverse implementation in a more general warping application (see Section V-F).

If forward implementation is used, the pixels of $O(x, y)$ may be computed by resampling the mapped pixels of $I(x, y)$ using some reasonable interpolation method, such as piecewise linear interpolation over the Delaunay triangulation of the mapped pixel locations $W(x, y)$. Interpolation in general introduces some amount of smoothing which is counterproductive for the present application. However the sampling rate is on the average 1/pixel and no smoothing in practice is encountered.

B. Solving the Differential Equation

As stated earlier, the warping function u is formulated as the solution (in each component) of a partial second-order elliptic differential equation. After discretization the equation

corresponds to a sparse linear system of order $N = \text{number of pixels in the image}$, with five nonzero entries in each row. Classical iterative solvers (Jacoby, Gauss–Seidel) prove notoriously slow in such situations, since their local action suppresses only high-frequency errors. However, in the present scenario pixel movement is quite restricted, and the initial guess of $u_{\text{init}}(x, y) = (mx, my)$ ($m = \text{scaling factor}$) satisfies $\|u_{\text{init}}(x, y) - u(x, y)\|_{\infty} < C$, with $C \approx 1$. Tests carried out so far show that under these assumptions the solution may be achieved in under 15 Gauss–Seidel iterations, thus the use of more advanced solvers is not necessary. If however the clamping parameter μ is set to zero, high values of λ give rise to strong warping effects (see Fig. 6), and the classic Gauss–Seidel iterative solver proves too slow. To accelerate convergence in such a case, a multigrid solver [22] was used. The number of multigrid iterations needed to achieve convergence is known to depend on the smoothness of the image-dependent component. When using “reasonable” values for λ , no more than seven to eight multigrid iterations were invoked, rendering the whole procedure (even in this case) realizable in relevant/real-time.

C. Constructing the Enhanced Image

The solution of the differential equation produces for each pixel of the output image the location (in the source image)



Fig. 6. The performance of the enhancer with the clamping parameter set to zero. Top: Original image. Middle: Result with $\lambda = 2$ as in Fig. 2. Bottom: Result with $\lambda = 7$. Such an effect may be useful in the entertainment industry for the creation of next-generation Vulcans.



Fig. 7. Output of the two additional flavors of the enhancement procedure. Top: Application of the warping low-pass filter on the original image. Middle: Result of applying (11) to the original image. Bottom: Result of applying (12) to the original image. Compare with Fig. 2. Note that as opposed to the original formulation, ringing appears, especially with the application of (11). Original image appears in Fig. 2(a).

from which the RGB values should be taken. These locations however are, in general, of noninteger coordinates and an approximation scheme is called for. We have chosen to approximate RGB values at noninteger positions by fitting a piecewise bilinear surface over the source image domain. A second possibility is to incorporate an unsharp masking component into the interpolant, a point we elaborate on in Section V. The bilinear interpolant potentially introduces some amount of smoothing, however due to the high sampling rate no smoothing is encountered in practice. In any case, other interpolants may be used, and the question of which technique to use in the resampling step is one of general importance (both in image-processing and image warping) and is not a focal point of the present work.

A demo package implementing the algorithm, written in C/C++ under the UNIX programming environment together with a Tcl/Tk interface may be found at http://www.cs.technion.ac.il/~gotsman/sharp_demo.tar.gz.

V. DISCUSSION

A. Comparison to Unsharp Masking

A connection between our method and unsharp masking may be found in the following: One may formulate the unsharp mask filter as

$$f \mapsto (1 + \mu)f - \mu LPS(f), \quad \mu > 0 \quad (10)$$

where $LPS(f)$ is a low-pass filtered copy of the original signal f . When using linear techniques, the term “lowpass filtered copy” has a precise and well-studied interpretation. However, as Kresch has so appropriately pointed out, one may interpret this term as the output of nonlinear filters [23], with diverse enhancement schemes resulting.

In our framework, one may formulate two additional ‘flavors’ of the proposed enhancement scheme as

$$f \mapsto (1 + \mu)f - \mu \mathcal{B}(f) \quad (11)$$

or

$$f \mapsto (1 + \mu)f - \frac{1}{2} \mu (\mathcal{S} - \mathcal{B})(f) \quad (12)$$

where \mathcal{S} is the enhancement operator described throughout this work, and \mathcal{B} is a *blurring* operator, computed in the same manner as \mathcal{S} , but with the image-dependent component taken as the reciprocal of the image-dependent component used for \mathcal{S} . \mathcal{S} may be interpreted as a “warping high-pass filter,” and \mathcal{B} is interpreted as a “warping lowpass filter.” Since the computation of the warping operator consists the main body of computations needed for the procedure (when $m = 1$), only one of \mathcal{B} and \mathcal{S} is computed, and the second is obtained from the first by the following procedure (shown only for the trivial scaling factor $m = 1$): Supposing, for example, that $\mathcal{S}: x \rightarrow x + \delta(x)$, then \mathcal{B} is approximated by $\mathcal{B}: x \rightarrow x - \delta(x)$. The validity of this procedure follows from the fact that, in practice, $\delta \approx 0$, and thus $\delta(x)$ may be well approximated by its first-order Taylor expansion. An example of these flavors of the procedures is shown in Fig. 7.

B. Comparison to the Perona–Malik Method

The enhancement method of Perona and Malik [14], modifies the diffusion equation in order to sharpen the inter-region edges while smoothing out homogeneous regions. The same type of effect is generally exhibited by the procedure presented in this paper. We, however, identify one drawback of the Perona–Malik method relative to ours: Anisotropic diffusion smoothes out edges whose “strength” is below a certain threshold, regardless of their “true” importance. Weak edges will be blurred even if they dominate their neighborhood. This is true even if the parameters driving the diffusion process are computed per image. On the other hand, sharpening by warping enhances any edge provided it dominates its neighborhood. The exact perceptual meaning of “dominates” is beyond the scope of this research, but in the framework of sharpening by warping its mathematical meaning is clear-cut. An example of these remarks is shown in Fig. 8.

The small perceptual similarity between the methods notwithstanding, a detailed mathematical comparison is beyond our reach; the Perona–Malik method modifies pixel intensities, while enhancement by warping modifies pixel positions, and a technique to compare some of the analytic properties of these two methods is yet to be found.

C. Color Images

Color images may be dealt with in our framework in two distinct manners. This follows from the fact that the warp is image-dependent. In image-independent warping scenarios, one warp is generated from S , and this is applied to each of the color components independently. When performing image-dependent warping, a similar effect may be obtained by computing the warp on the combined scalar intensity image only, and applying it to each of the color components. The color images in this paper were all produced with one mapping function applied to the three color components. When applying traditional enhancement techniques to color images, a problem arises; processing each color plane independently gives rise to color-aberrations. To this end the RGB color space is transformed into another color space where one of the components is luminance, processing is performed only on the luminance component and the result is then transformed back to RGB space. Our method avoids the subtle question of exactly which color space to use for the processing stage.

D. Image Warping and Image Processing

Although widespread and easily implemented, linear image processing procedures aimed at sharpening and scaling all suffer from the same drawbacks: (1) They exhibit overflow effects which are solved by the application of nonlinear filters. (2) In many cases the filter design makes the tacit assumption that the signal is band-limited, an assumption known to be false in images that include sharp edges and discontinuities, in which case the spectrum has an exceptionally slow decay [24]. On the other hand, some linear filtering techniques are designed to enhance images in which the blurring process is mathematically modeled [25], [26], and as such may exhibit



Fig. 8. Application of anisotropic diffusion on the image shown in Fig. 4(c). All results were obtained with 20 iterations, $\lambda = 0.01$ and the transfer function $g(\cdot) = \exp(-(\cdot/K)^2)$. Left to right: $K = 8, 20, 40$, respectively. As may be seen, edges are enhanced or smeared according to the absolute (versus local relative) strength.

mediocre performance in general situations. The method we propose is not model based, thus does not make assumptions about the spectral characteristics of the image or about the reason enhancement is needed. As such it may perform ‘on the average’ quite well. Moreover, since the linear filtering and image warping enhancement methods operate on different attributes of the image (the first on pixel values, the second on their locations), they can be applied sequentially in order to mutually improve their effects. Mutual application of these methods may also be achieved in the following manner: Although the mapping is defined as the exact solution to a linear system of equations, the solution itself is obtained by iterative methods, with the initial solution being uniform scaling, and each iteration approaching the exact solution. Thus one may terminate the iterative process prematurely (e.g., after *one* Gauss–Seidel/multigrid iteration) and then enhance the result with linear filtering.

E. Enhancement of JPEG Compressed Images

A large portion of images encountered in digital applications are stored in the lossy JPEG-compressed format. JPEG compression with the quality parameter of 75% is considered in many cases as perceptually lossless, and indeed many high-quality images are compressed using this value. Fine examination of these images (sometimes with the aid of nonlinear lookup tables) reveals that JPEG artifacts, namely the 8×8 blocks, are present. When applying an unsharp masking filter to these images the blockiness is amplified, and in many cases is visible even at a first glance. Thus, assuming no JPEG postprocessing is applied to the image, these filters may prove counter-productive. We speculate that the same is true of any enhancement method (e.g., morphological filtering) which amplifies differences in RGB values of neighboring pixels. For example, traditional enhancement methods applied to images scanned from medium to low-grade paper result in amplification of the paper texture. Since the method we propose modifies pixel location rather than values, it may be better suited for such a setting, since it guarantees not to amplify additive noise. Application of the two enhancement methods on a lossy compressed image is shown in Fig. 9.

F. Minimal-Distortion Warping

Another application in which image-dependent warping may yield results superior to those of existing methods, is minimal-distortion warping. This is actually the main focus in many image warping applications. In this case, a warp is specified by S on an image I , and the problem is to find the warp W_S which obeys S , yet distorts I as little as possible. The term “distorts” is not well-defined, and has been given different meanings by various authors, usually the value of a functional such as (1). However, it is obvious that a geometric mapping $W: \mathbb{R}^2 \rightarrow \mathbb{R}^2$ applied to one image may yield a result appearing very distorted by all definitions, and on another image not distorted at all. This is because the visual effect of a local warp of the plane depends on the image activity (the amount of visual information) in that locale. If the image is relatively flat and smooth, the underlying geometric distortion will be noticed much less than if the image is very active. So, if we are to talk about *image* distortion, versus geometric distortion, I must be taken into account by weighting the geometric factor according to image activity. In minimal-distortion warping, the main idea is to “sweep the distortion under the carpet” in a sense, namely, geometrically distort the areas of the plane in which it will be less likely to be noticed. Areas of the image which are more active are then less likely to be visually distorted.

An image-dependent regularization functional inspired by the thin-plate functional whose minimization could achieve our aim is the following:

$$F(I) = \iint_{\mathbb{R}^2} (u_{xx}^2 + 2u_{xy}^2 + u_{yy}^2)(1 + \lambda|\nabla I|) dx dy. \quad (13)$$

The image-independent component (the thin-plate functional) measures the nonlinearity of the mapping, since it is non-negative and vanishes on linear terms, while the image dependent component forces the nonlinear effects to concentrate in the smooth parts of the image, where they are much less noticeable. Unfortunately, minimizing $F(I)$ subject to data interpolation even in the 1-D case is quite involved [27], and a closed form solution in the bivariate case is not known. To this end we propose to numerically minimize the same integral taken over the domain for which the image is defined. The problem then reduces to the solution of a square sparse



Fig. 9. Enhancement of a JPEG-compressed image. (a) Original image. (b) Unsharp masking applied to original image, after bilinear scaling $\times 4$. (c) Enhancement and scaling $\times 4$ by proposed method. All images are shown at same size for presentation purposes only. Original image is compressed with quality 75%. The sharpening parameter used in the unsharp masking is 75%, a commonly used value. The graininess of the right image is due to the interaction of three components of the printer pipeline: Lossy JPEG compression, unsharp masking and printer halftoning. (d) Enlargement of selected area of (b) showing enhanced JPEG artifacts.

linear system of order $N = \text{number of image pixels}$, and as in Section II-B, highly efficient numerical procedures may be invoked. We note that in the case $1 + \lambda|\nabla I| \equiv \text{const}$, the problem reduces to that of minimizing the thin-plate integral over a bounded domain [28]. Furthermore the whole procedure is affine-reducible as in the classical thin-plate spline case. The interpolation constraints driving the warp may be either point-to-point, as in the classical thin-plate spline case, or boundary conditions similar to the ones present in the image-enhancement application presented here.

The possibility of applying this method in order to obtain improved results in classical warping settings is still under investigation. Note that as opposed to the enhancement application presented, pixel movement is not bounded, and thus a multigrid PDE equation solver may prove instrumental in obtaining a solution in reasonable time. Moreover, since backward implementation should compute the image-dependent component from the target image, such an implementation is not possible for such a setting.

G. Future Work

The most serious drawback of the method presented is its global nature. This drawback hinders performance in two

ways: Computational complexity (compared to local filtering) and global effects. These may be easily seen in Fig. 6, where the clamping parameter is set to zero, and over-enhancement leads to distortion. Additional reduction of global effects may be achieved by first segmenting the image and then applying the proposed enhancer on each segment independently. Such preprocessing may also be applied to the enhancement of text images where segmentation is quite easily achieved.

When suppressing the global effects we have chosen to use the same image-dependent component as the one used for the edge-enhancement. This by no means is the only possibility. One may, in a more general setting formulate the mapping as the solution of

$$\min_{u(x,y)} \iint_{[0,1]^2} ((u_x^2 + u_y^2)\mathcal{I}_1(x,y) + (u(x,y) - m(x,y))^2 \cdot \mathcal{I}_2(x,y)) dx dy \quad (14)$$

with \mathcal{I}_1 and \mathcal{I}_2 being two image-dependent components. For instance \mathcal{I}_2 can be taken as a thresholded version of the edge-detector \mathcal{I}_1 , with values above threshold set to $+\infty$, giving rise to a segmentation procedure.

Efficient Implementation: The mapping function is obtained as the solution to a partial differential equation whose domain is the image plane. Thus at least on a theoretical level, implementing the procedure by buffering up only $N \ll$ (height of image) scanlines is not possible. However, in practice, if the parameters λ and μ are properly tuned, the effect of the mapping is highly localized [as is also suggested by Fig. 2(f)], and scanline implementation may be possible. We intend to further pursue this issue.

REFERENCES

- [1] J. Flusser, "An adaptive method for image registration," *Pattern Recognit.*, vol. 25, pp. 45–54, 1992.
- [2] T. Beier and S. Neely, "Feature-based image metamorphosis," *Comput. Graph.*, vol. 26, pp. 35–42, 1992, also in *Proc. SIGGRAPH'92*.
- [3] S. Y. Lee, K. Y. Chwa, S. Y. Shin, and G. Wolberg, "Image metamorphosis using snakes and free-form deformations," *Comput. Graph.*, vol. 29, 1995, also in *Proc. SIGGRAPH'95*.
- [4] G. Wolberg, "Skeleton-based image warping," *Vis. Comput.*, vol. 5, pp. 95–108, 1989.
- [5] N. Arad, "Designing and implementing a grid-distortion mapping based on variational principles," *Comput. Graph. Forum*, vol. 13, pp. C-259–C-270, 1994, also in *Proc. Eurographics'94*.
- [6] N. Arad, N. Dyn, D. Reissfeld, and Y. Yeshurun, "Image warping by radial basis functions; application to facial expressions," *Graph. Models Image Process.*, vol. 56, pp. 161–172, Mar. 1994.
- [7] N. Arad and D. Reissfeld, "Image warping using few anchor points and radial functions," *Comput. Graph. Forum*, vol. 14, pp. 35–46, Mar. 1995.
- [8] D. Ruprecht and H. Muller, "Image warping with scattered data interpolation," *IEEE Comput. Graph. Applic.*, vol. 15, pp. 37–43, Mar. 1995.
- [9] P. S. Heckbert, "Survey of texture mapping," *IEEE Comput. Graph. Applic.*, vol. 6, pp. 56–67, Nov. 1986.
- [10] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 567–585, 1989.
- [11] I. Barrodale, D. Skea, M. Berkley, R. Kuwahara, and R. Poekert, "Warping digital images using Thin-plate splines," *Pattern Recognit.*, vol. 26, pp. 375–376, 1993.
- [12] P. Litwinowicz and L. Williams, "Animating images with drawings," *Comput. Graph.*, vol. 28, pp. 409–412, 1994, also in *Proc. SIGGRAPH'94*.
- [13] P. Borrel and A. Rappoport, "Simple constrained deformations for geometric modeling and interactive design," *ACM Trans. Comput. Graph.*, vol. 13, pp. 137–155, Apr. 1994.
- [14] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 629–639, 1990.
- [15] P. Zamperoni, "An adaptive rank filter for image enhancement depending on a measure of the local spatial order," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 9, pp. 127–144, 1995.
- [16] D. Reissfeld, *Generalized Symmetry Transforms: Attentional Mechanisms and Face Recognition*, Ph.D. dissertation, Tel Aviv Univ., Jan. 1994.
- [17] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 679–698, 1986.
- [18] Z. C. Li, C. Y. Suen, T. D. Bui, and Q. L. Gu, "Harmonic models of shape transformations in digital images and patterns," *Graph. Models Image Process.*, vol. 54, pp. 198–209, May 1992.
- [19] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, vol. 4 of *Studies in Mathematics and its Applications*. Amsterdam, The Netherlands: North-Holland, 1978.
- [20] G. Strang, *Introduction to Applied Mathematics*. Wellesley, MA: Wellesley-Cambridge Press, 1986.
- [21] S. K. Parke and R. A. Schowengerdt, "Image reconstruction by parametric cubic convolution," *Comput. Vis. Graph. Image Process.*, vol. 23, pp. 258–272, 1983.
- [22] Y. Shapira, M. Israeli, and A. Sidi, "Toward automatic multigrid algorithms for SVD, nonsymmetric and indefinite problems," *SIAM J. Sci. Comput.*, vol. 17, pp. 439–453, 1996.
- [23] R. Kresch and G. Sapiro, "Morphological image sharpening," Tech. Rep. HPL-96-67, Hewlett-Packard Labs., May 1996.
- [24] Y. Katznelson, *An Introduction to Harmonic Analysis*, 2nd ed. New York: Dover, 1976.
- [25] M. Lindenbaum, M. Fischer, and A. Bruckstein, "On Gabor's contribution to image enhancement," *Pattern Recognit.*, vol. 27, pp. 1–8, 1994.
- [26] P. W. Wong and C. Herley, "Area-based interpolation for scaling of images from a CCD," Hewlett-Packard Labs. Internal Memo, 1995.
- [27] F. Abramowitz, "Spline smoothing with variable smoothing parameter," Ph.D. dissertation, School Math. Sciences, Tel-Aviv Univ., Aug. 1992.
- [28] N. Dyn and D. Levin, "Construction of surface spline interpolants of scattered data over finite domains," *RAIRO Numer. Anal.*, vol. 16, pp. 199–209, 1982.

Nur Arad received the B.Sc. and M.Sc. degrees in mathematics and the Ph.D. degree in mathematics and computer science, all from Tel Aviv University in 1984, 1987, and 1995, respectively.

From 1995 to 1996, he was with the Hewlett-Packard Israel Science Center, Haifa, Israel. He is currently with the Algorithms Group at Orbotech Ltd, Yavne, Israel. His research interests include computer graphics, CAGD, and image processing.

Craig Gotsman received the B.Sc. degree in mathematics, physics and computer science in 1983, the M.Sc. degree in computer science in 1985, and the Ph.D. degree in computer science in 1991, all from the Hebrew University of Jerusalem, Jerusalem, Israel.

He is a Senior Lecturer with the Department of Computer Science, Technion-Israel Institute of Technology, and currently on sabbatical leave at Virtue Ltd, Haifa, Israel. His research interests include computer graphics, image rendering, and geometric modeling.