

The Multiresolution Gradient Vector Field Skeleton

Wooi-Boon Goh^{a,*} and Kai-Yun Chan^b

^a School of Computer Engineering,
Nanyang Technological University,
Nanyang Avenue
Singapore 639798

^b Wycliffe Singapore
34 Craig Road
#B1-06/07 Chinatown Plaza
Singapore 089673

* Corresponding Author:
Email: aswbgoh@ntu.edu.sg
Phone: 65-67904611
Fax: 65-67926559

Abstract

Many algorithms suppress skeleton associated with boundary perturbation by preventing their formation or by costly branch pruning. This work proposes a novel concept of structural and textural skeletons. The former is associated with the general shape structure and the latter with boundary perturbations. These skeletons remain disconnected to facilitate gross shape matching without the need for branch pruning. They are extracted from a multiresolution gradient vector field that is efficiently generated within a pyramidal framework. Experimental results show that these skeletons are scale and rotation invariant. They are less affected by boundary noise compared to skeletons extracted by popular iterative and non-iterative techniques.

Keywords: Skeleton, Shape Description, Medial Representation, Multiresolution

1. Introduction

Skeletons have been widely adopted as a basis for shape description [10], [14], [22], [23], [28]. This form of representation is popular because it is highly amenable to part-based decomposition of shapes. Furthermore, Kimia [11] recently highlighted several psychophysical and neurophysiological evidences that suggest some role for medial axis in the human visual system. Despite these facts, it is still a challenge to extract robust skeletons in the presence of boundary noise. A popular definition adopted to define a shape's skeleton is the locus of centers of maximal disks contained within the shape [5]. Such skeletons allow the original shape to be fully reconstructed if the radii of the disks are associated with each skeletal point. The instability of such skeletons is well-known since small perturbations at the boundary can significantly alter the structure of the skeleton [3]. An alternative approach to using maximal disks is to extract the skeleton by tracing out the singular points of an advancing 'grassfire' front started simultaneously at all points on the boundary. Such singular points are defined as points where the fronts meet. They are well-defined at points where the fronts collide from different directions but become numerically more difficult to determine at intersections where the angular differences between the colliding fronts are small, especially when such computations are done within discrete grids. A distance function D can be derived within the shape by noting the time of arrival of the front at every point and this in essence is similar to the Euclidean distance function to the boundary [7]. Detecting the local maxima on the surface of the distance function will also yield the skeleton of the shape [2].

The sensitivity of skeleton extraction algorithms to boundary noise is notorious (see Fig. 1a) and has been traditionally addressed using some form of pre-smoothing technique or branch pruning strategy. In the case of the former, pre-smoothing the shape contour using curvature flow smoothing [10], [17] techniques could be carried out before extracting the skeleton. Alternatively, Tari & Shah [31] propose smoothing the distance function D within the shape using constraints that will ensure its value remains zero at the boundary. A more popular approach is to allow the noise induced skeleton to form and then apply some pruning strategy to remove insignificant branches. Shaked and Bruckstein [25] proposed several topology preserving pruning methods that are continuous in its threshold parameter. Multiscale pruning strategies such as the iteratively pruned trees of Voronoi edges have also been proposed [19]. Shah [24] argued that the

problem with most skeletonization algorithm is that they make binary decisions about whether a point belongs to the skeleton. He proposed using the angle between the grassfire front normal as a normalized monotonic measure to describe the probability a point within a shape is part of what he termed the *gray skeleton*. An application dependent threshold could be later used to set the local context for pruning inconsequential skeletons. The disconnected skeletons are subsequently connected by extending their branches towards the shape’s interior along with some additional constraints. A novel multiscale approach for tackling the noise problem was proposed by Pizer et al. [20]. They described the medialness function (cores) of a shape over a width-proportional scale. The cores describing the interior medialness of shapes can be extracted in the wider scales and cores describing minor boundary protrusions and perturbations can be extracted at finer scales. By selecting cores from appropriate scales, core-based analysis of shapes can be made relatively insensitive to boundary perturbation [18]. In summary, most of these approaches seek to suppress spurious skeletons either by preventing their formation or by removing them subsequently. However, it has been acknowledged that selecting an appropriate scale or pruning threshold that can properly define what constitutes an insignificant skeleton is non trivial [21]. Moreover, spurious skeletons may be useful in describing meaningful boundary perturbations such as the serrated teeth patterns that distinguishes different keys (see Fig. 1).

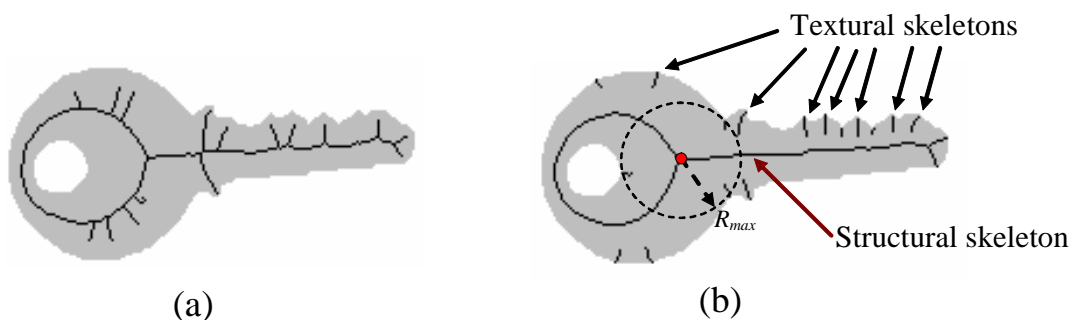


Fig. 1. (a) Sensitivity of a homotopic skeleton to boundary perturbation. The skeleton was extracted from the Euclidean distance transformed surface. (b) The proposed structural and textural skeletons. The structural skeleton represents the general topology of the shape, whilst the disconnected textural skeletons capture prominent boundary artifacts, such as the serrated teeth pattern on the key. The structural skeleton contains the center of maximal disk with the largest radius R_{max} .

This paper is organized as follows. Section 2 proposes our solution to this problem, by way of using disconnected structural and textural skeletons that are extracted from the multiresolution gradient vector field (MGVF) introduced in section 3. Section 4 describes a scale-invariant scalar function in the form of a vector field directional disparity map and algorithms for extracting the proposed skeletons. Section 5 presents a series of experimental results to highlight the characteristics of the MGVF skeleton and compares its performance with several well known skeletonization algorithms. Their relative computational requirements and characteristics are also discussed.

2. Structural and Textural Skeletons

A more appropriate approach is to view a shape as comprising two types of skeletons, namely structural and textural skeletons (see Fig. 1b). In this work, the *structural skeleton* is defined as the fully connected skeletal segment containing the skeletal point that is the center of the largest maximal disk with radius R_{max} . This definition results in a single connected skeleton that describes the general structure and topology related to the widest area within the shape (see Fig. 1b). This structural skeleton is relatively stable to boundary perturbation. Alternative definitions that are less rigid may also be adopted. For example, structural skeletons can be defined as connected skeletal segments that contain any skeletal point whose maximal disk radius is greater than some predefined threshold such as $0.5 \times R_{max}$. This definition may be more suitable for shapes such as that shown in Fig. 2a as it allows for multiple disconnected structural skeletons.

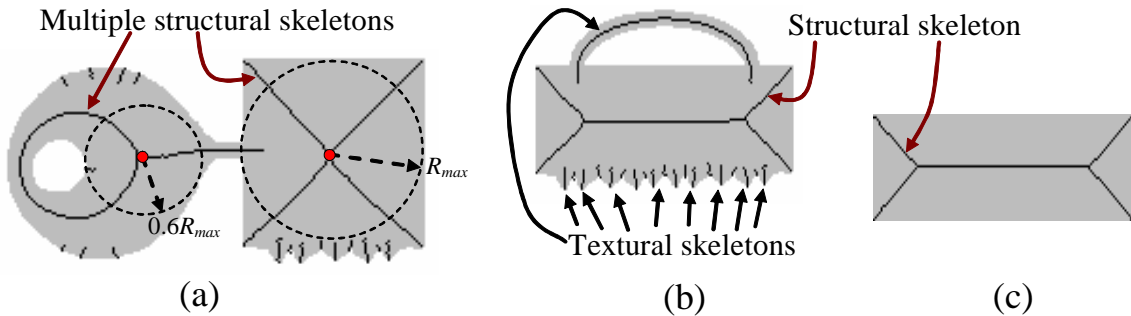


Fig. 2. (a) A maximal disk radius threshold-based approach for extracting multiple structural skeletons. (b) Textural skeletons may also have topological properties. (c) Defining structural skeletons using a widest-region criterion allow them to represent the gross shape of an object based on salient visual regions and not shape topology.

Textural skeletons on the other hand, are more loosely defined as all remaining skeletal segments that are not part of the structural skeleton. These skeletal segments are normally associated with boundary perturbations and they emanate from the boundary towards the interior (see Fig. 1b). This dual skeleton notion was also highlighted by Pizer et. al in [21]. They used the term *topological skeleton* to refer to the part of a skeleton which cannot be further contracted without altering its corresponding topology. Skeletal branches emanating from the topological skeleton towards the boundary (see Fig. 1a) define finer shape details and can be deleted without affecting topological characteristics. However, these skeletons are usually connected by virtue of the way they are constructed [13], [32] or in some cases where they were not [24], [29], homotopy preserving algorithms were then employed to ensure connectivity is maintained. With such skeletal representation, some form of pruning strategy must be employed to remove insignificant branches before the ‘noise-free’ skeleton can be extracted for subsequent shape analysis. Since we have adopted a region width criterion (in the form of maximal disk radius) to define structural skeletons, it is quite possible that narrow topologically-related parts of a shape are classified as textural skeletons (e.g. curved handle in Fig 2b). However, it can be argued that since the limb-like parts are narrow, it does not detract from the fact that the general perceived structure of the shapes in Fig. 2b and 2c (based of salient visual regions) are both rectangles, as reflected in their similar structural skeletons. In the context of this work, only the additional inclusion of some length-based criterion would allow the skeleton of the narrow curved handle of Fig. 2b to acquire structural status.

We argue that the structural and textural skeletons should be encouraged to remain disconnected (see Fig. 1b). In this way, both the gross shape structure and boundary perturbations can be simultaneously represented and accessible without the need for costly branch pruning. This seems to run contrary to the homotopic nature of most skeletonization algorithm. A fully connected skeleton is useful because it can be represented using a single graph and this allows shapes to be matched using well-studied graph matching algorithms [14], [23], [28]. However, under noisy conditions, homotopic skeletons can result in dense graphs that contain many spurious branches. This makes subsequent graph matching complicated and timing consuming. For example, the edit distance algorithms of Sebastian et. al [23] is able to prune away a leaf edge in the shock graph and associate an appropriate cost for such a splice operation based on the significance of the resulting boundary changes. But pruning the numerous spurious leaf edges arising from a shape with high boundary noise can significantly increase the time

taken to compute the optimal match between two shapes. However, if the relatively boundary noise-free structural skeletons are disconnected from the textural skeletons (see Fig. 3b), they can be quickly compared to determine their respective gross shape matches before textural skeletons are employed to refine the match scores among the top few candidates. Disconnected skeleton-based descriptions of shape have also been adopted in the previous works, albeit for other reasons. Rom and Medioni [22] focused on the hierarchical decomposition of shapes. In their work, ribbon-like parts defined using smooth local and parallel symmetries were used to extract global relationships within the shape and for defining the hierarchical nature of the parts within the shape. In the case of Liu and Geiger [14], their work was driven by their need to describe shapes with articulating limbs (e.g. humans). As such, disconnected ribbons were formed by identifying segments of the shape’s boundary that have a close mirrored counterpart (e.g. limbs). These disconnected shape axes are then connected graphically to form an SA-tree, which was then used for shape matching.

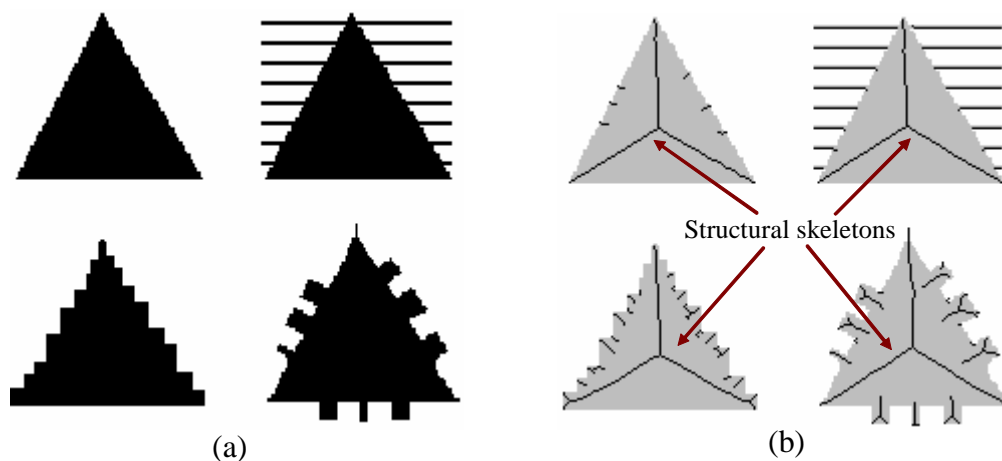


Fig. 3. (a) Four triangularly shaped objects with varying boundary characteristics and (b) their respective disconnected structural and textural skeletons. Gross classification of shapes can be made by just by comparing their structural skeletons.

3. Multiresolution Gradient Vector Field

Several researchers have proposed the use of vector fields to analyze symmetry in shapes. Cross and Hancock [9] proposed a multiscale framework where a vector field is obtained by computing the curl of vector potential found through volume averaging of the tangential edge gradient vectors. Symmetry axes were represented by locations where the

curl of the vector potential vanishes. These symmetry points were extracted by fitting a local quadratic prism surface to the magnitude of the vector potential in each 3×3 image neighborhood and then applying a non-minimum suppression test to detect directionally consistent symmetry features. Another multiresolution skeletonization algorithm based on the Electrostatic Field Theory (EFT) was proposed by Abdel-Hamid et al. [1]. The EFT-based method provided skeletal representation across a variety of scales by changing the starting potential value during skeleton generation. Shroff and Ben-Arie [27] modeled the boundary gradient of a shape as magnetic dipoles and extracted disconnected skeletons (shape axes) at point where the magnetic field interaction resulted in a local minima. Their shape axes were relatively smooth and insensitive to noise. In our work, a gradient vector field is also used as a basis for generating a scalar 2D surface, from which the skeleton can be extracted by detecting the local maxima on this surface. A computationally efficient pyramidal framework is proposed for generating a gradient vector field that exhibits characteristics which facilitates the extraction of a disconnected boundary noise-robust structural skeleton. Integration of gradient information over different spatial resolutions within the pyramid allows the gradient vectors in the interior of the shape to be constituted with noise-suppressed low spatial frequencies. Whilst, vectors near the boundary are constituted with detail-preserving high spatial frequencies.

In order to describe the pyramidal framework used to generate the MGVF, we first review several pyramid operations. The process which generates a lower resolution image from its predecessor will be called a *REDUCE* operation [8]. If G_0 is the original image $I(x, y)$ and G_N is the top level of the pyramid, then for $0 < l \leq N$, $G_l = \text{REDUCE} [G_{l-1}]$ is define as

$$G_l(x, y) = \sum_{m=-k}^k \sum_{n=-k}^k w(m, n) G_{l-1}(2x + m, 2y + n) \quad (1)$$

where the generating kernel w of size $(2k+1)$ performs smoothing before the sub-sampling process. For computational efficiency reasons, a separable, normalized and symmetric Gaussian kernel of $k = 2$ was used throughout [8]. Another pyramid operation, *EXPAND* is used to expand an image of size $M+1$ to $2M+1$ by interpolating sample values from a low resolution image. If G_l is derived by expanding it low resolution image G_{l+1} , then for $0 \leq l < N$, $G_l = \text{EXPAND} [G_{l+1}]$ is define as

$$G_l(x, y) = 4 \sum_{m,n=-k}^k w(m,n) G_{l+1}\left(\frac{x+m}{2}, \frac{y+n}{2}\right) \quad (2)$$

where summation is only carried out when $x+m$ and $y+n$ are even numbers.

The steps required to generate the gradient vector field pyramids is summarized in Fig. 4. It is assumed that the input image is a binary silhouette image $I(x, y)$. Firstly, a Gaussian pyramid $G(l, x, y)$ of $N+1$ levels is created by iteratively applying the *REDUCE* operation to each consecutive output image, starting with the original image $I(x, y)$. From the scalar Gaussian pyramid, the vectorial Gradient pyramid $\mathbf{H}(l, x, y)$ is then derived by applying two derivative filters to every image pixel in the Gaussian pyramid. The resulting Gradient pyramid consist of two scalar pyramids $H^x(l, x, y)$ and $H^y(l, x, y)$ given by

$$\begin{aligned} H_l^x(\mathbf{x}) &= g_{\sigma_H}^x(\mathbf{x}) * G_l(\mathbf{x}) \quad \text{and} \\ H_l^y(\mathbf{x}) &= g_{\sigma_H}^y(\mathbf{x}) * G_l(\mathbf{x}) \quad \text{for } 0 \leq l \leq N \end{aligned} \quad (3)$$

where $\mathbf{x} = (x, y)$ and $*$ is the convolution operator. The convolution kernels are Gaussian derivatives of the first-order in the x and y directions, respectively and are given by

$$g_{\sigma_H}^x(\mathbf{x}) = -\frac{x}{\sigma_H^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_H^2}\right) \quad \text{and} \quad g_{\sigma_H}^y(\mathbf{x}) = -\frac{y}{\sigma_H^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_H^2}\right) \quad (4)$$

The Gradient pyramid essentially describes the gradient of the image $I(x, y)$ at octave scales as the pyramid is traversed from one level to another. The object's interior at lower levels of the Gradient pyramid usually has undefined gradient values, as can be seen in Fig. 4. This is because the size of the derivative kernels in (4) are intentionally kept small (typical size of 7×7 pixels) to reduce computational cost. A third pyramid called the Gradient Vector Field pyramid is then constructed to provide a dense gradient vector field that is properly defined at all localities within the shape and at all image levels. It is constructed by iteratively applying the *EXPAND* operation to a proportioned sum of the corresponding level \mathbf{H}_l of the Gradient pyramid and the result of the previous iteration given by $EXPAND[\mathbf{V}_{l+1}]$, as illustrated in Fig. 4. The apex of the pyramid, \mathbf{V}_N is first initialised with the value of \mathbf{H}_N , the apex of the Gradient pyramid. This is necessary so that the iterative expand and summate process can be subsequently initiated, starting at $l=N-1$ and terminating at $l=0$. More formally, this *expand-summate* process is defined as

$$\begin{aligned}
V_l &= H_l && \text{for } l = N \\
V_l &= \alpha H_l + (1-\alpha) \text{EXPAND}[V_{l+1}] && \text{for } 0 \leq l < N
\end{aligned} \tag{5}$$

A smoothing parameter $\alpha \in [0, 1]$ is introduced to govern the summation ratio between one level and the next and its value determines the smoothness of the gradient vector field. A smaller α value results in a smoother vector field in the interior of the shape since the gradient vector field is constituted using a higher proportion of low spatial frequency components.

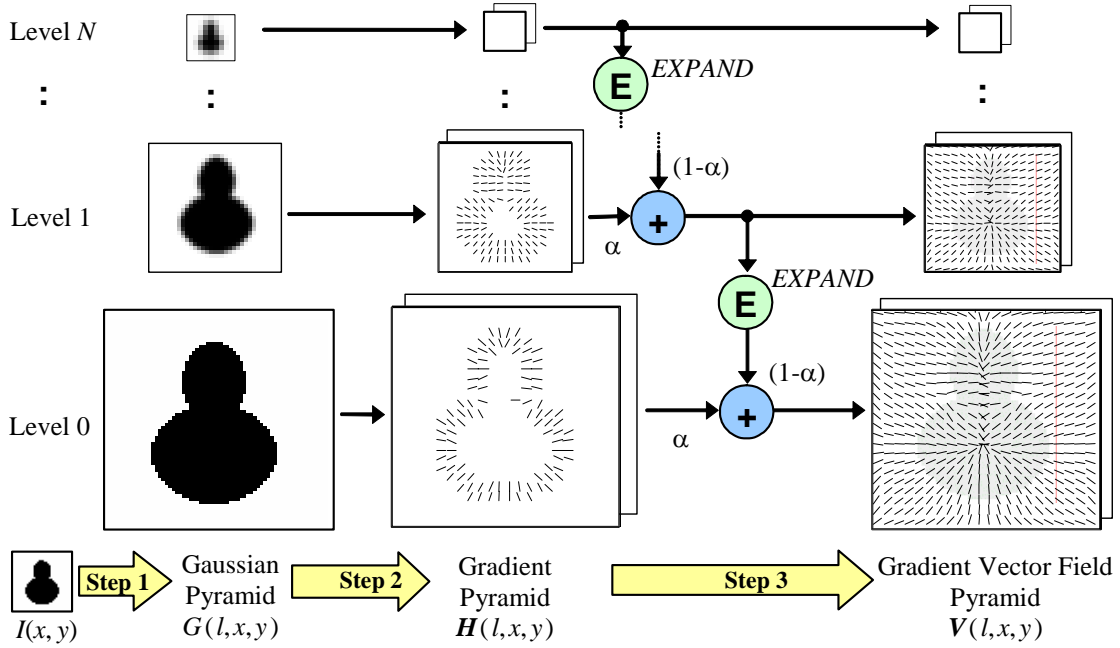


Fig. 4. The steps involved in constructing the Gradient Vector Field pyramid.

4. The MGVF Skeleton

The property of the extracted skeleton varies with the skeletonization algorithm employed and some properties may be fundamentally at odds with each other [5]. For example, skeletons that have high noise immunity are usually smooth and not plagued by undesirable perturbation-induced runners. However, such skeletons violate the mediality requirement, which dictates that skeletons must be jagged to accommodate boundary perturbations so that all points on the skeletons are equidistant from its two closest boundary points. What constitute a desirable property is ultimately dependent on the nature of the application. The MGVF skeleton was originally designed for use in part-based description of noisy shapes, with specific application in shape classification and

content-based image retrieval. Properties such as noise immunity, scale and rotation invariance exhibited by the MGVF skeleton are generally considered worthy goals for such purposes. Another characteristic the MGVF skeleton shares with the medial representation such as the ‘cores’ of Pizer et al. [20] is the disconnected nature of its skeleton (see Fig. 1b). It has been argued that this is a desirable feature for noise robust medial representation. The rest of this section details the procedure for extracting the skeleton from the MGVF introduced in the previous section. Unlike most existing skeletonization algorithms [13], [29], the extraction algorithm proposed here is specifically designed such that it does not enforce connectivity. However, some additional steps have been included to ensure that connectivity is maintained in the structural skeleton.

4.1 Vector Field Directional Disparity Map

Given the gradient vector field $V_l(x,y)$ in (5), the shape axes can be located by detecting locations in the vector field where the local gradient vectors exhibit high directional disparity. This is similar in concept to the technique proposed in [29], where Siddiqi et al. used the measure of average outward flux over a very small neighborhood to detect medial points. In this work, the idea of Ben-Arie and Wang in [4] is extended to give an alternative local directional disparity operator Δ given by

$$\Delta(\mathbf{v}(\mathbf{x}), \sigma_D) = \frac{g_{\sigma_D}(\mathbf{x}) * \|\mathbf{v}(\mathbf{x})\| - \|g_{\sigma_D}(\mathbf{x}) * \mathbf{v}(\mathbf{x})\|}{g_{\sigma_D}(\mathbf{x}) * \|\mathbf{v}(\mathbf{x})\|} \quad (6)$$

where $\mathbf{x} = (x,y)$ and the normalized local disparity measure $\Delta(\mathbf{v}(\mathbf{x}), \sigma_D) \in [0,1]$ is computed within a weighted locality defined by the Gaussian convolution kernel $g_{\sigma_D}(\mathbf{x})$ given by

$$g_{\sigma_D}(\mathbf{x}) = \frac{1}{\sqrt{2\pi}\sigma_D} \exp\left(-\frac{x^2 + y^2}{2\sigma_D^2}\right) \quad (7)$$

The local directional disparity operator Δ gives a value close to 1 at the center of a converging gradient vector field, such as one obtained at the center of a circle and a value close to 0 in a locality where the gradient vector field is unidirectional, as shown in Fig. 5.

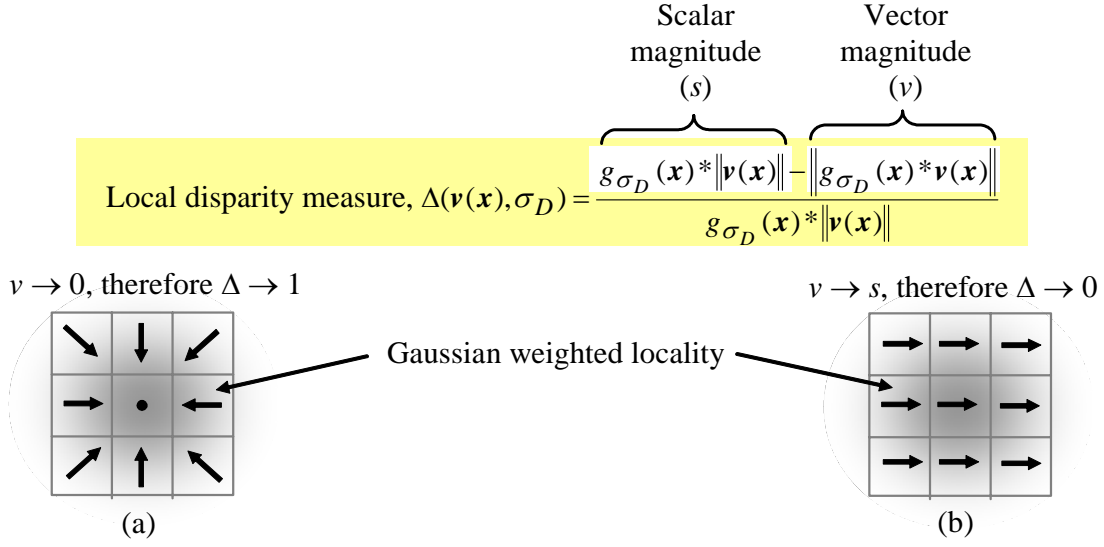


Fig. 5. The output of the local directional disparity operator Δ in different vector fields. (a) In a converging vector field, $\Delta \rightarrow 1$ and (b) in a unidirectional vector field, $\Delta \rightarrow 0$.

By applying the local directional disparity operator Δ to every level of the Gradient Vector Field pyramid $V(l, x, y)$, the Vector Field Disparity pyramid $D(l, x, y)$ is obtained, where each level D_l is given by

$$D_l(x, y) = \Delta(V_l(x, y), \sigma_D) \quad \text{for } 0 \leq l \leq N \quad (8)$$

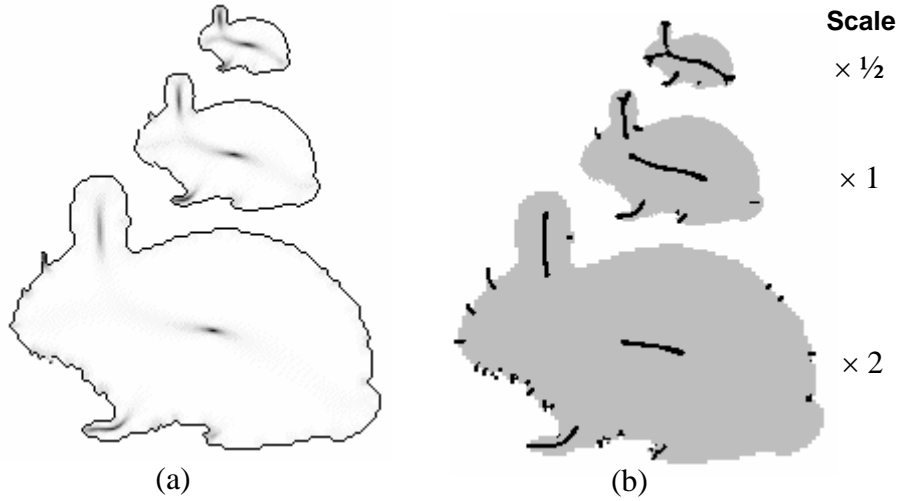


Fig. 6. Inconsistent shape axes extracted from a single level of the vector field disparity pyramid. (a) Disparity measure at level D_0 . (b) The local maxima ridges extracted from the D_0 using a simple local ridge detection algorithm similar to [27]. It flags out a ridge pixel when N_D of its neighboring pixels have disparity values less than its own. Results shown were obtained using $\alpha = 0.8$, $\sigma_D = 1.5$, a square 5×5 neighborhood and $N_D = 14$.

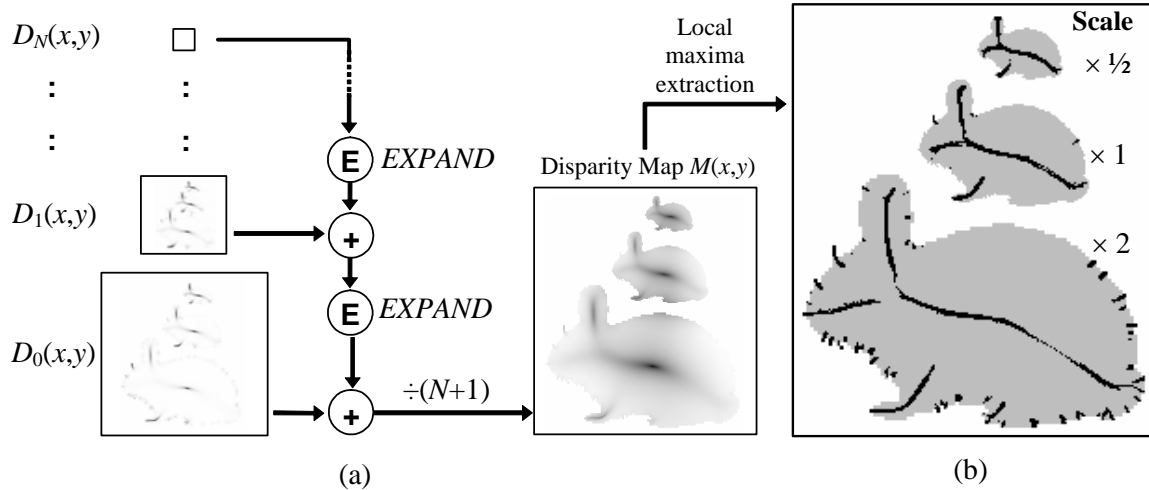


Fig. 7. (a) The multi-level integration technique for generating the vector field disparity map $M(x,y)$. (b) The local maxima ridges extracted from $M(x,y)$ with $N_D = 14$. Ridges at the boundary of the ($\times 2$) rabbit are due to discrete image zooming artifacts, which becomes sufficiently prominent at this scale and will be extracted.

Fig. 6a shows the disparity measure at the lowest level D_0 of an image containing three varying-sized rabbit shapes that span scales of two octaves and Fig. 6b shows the corresponding extracted local maxima. Notice that the extracted shape axes are not scale-invariant. Pizer et al. in [20] addressed this problem by extracting relevant ‘cores’ of shapes and its subparts of varying width while tracking optimal scale ridges over a 3D scale-space volume. This approach is computationally costly and may require fine scale resolutions. We propose a simpler solution, which basically involves integrating the different Vector Field Disparity pyramid levels into a single full resolution disparity map. This allows simultaneous extraction of all relevant medial axes across different scales using only a single scalar surface and simple local maxima detection algorithms. A full resolution vector field directional disparity map $M(x,y)$ is obtained by iteratively applying the $EXPAND$ operation to the sum of D_l and $EXPAND [D_{l+1}]$, starting at $l=N-1$ to $l=0$ (see Fig. 7). The final summed output is divided by $N+1$ to re-normalize the disparity map such that $M(x,y) \in [0, 1]$. In summary $M(x,y)$ is given by

$$M(x,y) = \frac{1}{N+1} \sum_{l=0}^N d_l(x,y) \quad (9)$$

where $d_l(x,y) = D_l(x,y)$ for $l = N$

$$d_l(x,y) = D_l(x,y) + EXPAND[d_{l+1}(x,y)] \quad \text{for } 0 \leq l < N \quad (10)$$

This process integrates what are essentially multi-scaled features spread over all levels of the Vector Field Disparity pyramid into a single full resolution disparity map, $M(x, y)$. The shape axes extracted in Fig. 7b are more scale invariant than those in Fig. 6b.

4.2 Extracting the MGVF Skeleton

Given a disparity map $M(x, y)$, the shape axes can be extracted by locating the local maximas in the disparity measure using a local ridge detection algorithm. However, the shape axes near bifurcation points were not properly extracted (see Fig. 7b). This is because the ridges in the interior of the shape have been significantly blurred due to the multiple *EXPAND* operations used to derive $M(x, y)$. This poses difficulty in the ridge extraction process. A more robust method of extracting shape axes from the disparity map is proposed. The first step involves computing the derivative of $M(x, y)$ given by

$$\mathbf{M}_V(x, y) = \nabla g_{\sigma_V} * M(x, y) \quad (11)$$

where ∇g_{σ_V} produces first-order Gaussian derivative convolution kernels with a standard deviation of σ_V and the resulting disparity gradient map \mathbf{M}_V is a vector consisting of the x and y derivatives of the disparity map given by M_V^x and M_V^y respectively. The disparity operator Δ in (6) is then applied to \mathbf{M}_V to obtain an enhanced vector field disparity map $M_E(x, y)$ that is shown in Fig. 8b and is defined by

$$M_E(x, y) = \Delta(\mathbf{M}_V, \sigma_E) \quad (12)$$

where σ_E determines the size of the neighborhood in which the directional disparity is computed. Unfortunately, the disparity operator Δ does not differentiate between ridges and valleys. Since gradient vectors of \mathbf{M}_V around ridge points are outward flowing and those around valleys are inward flowing, a landscape map $M_L(x, y)$ that highlights ‘ridgeness’ and ‘valleyiness’ in the disparity map can be easily constructed. This is done by finding the sum of the resulting two convolutions of M_V^x and M_V^y in (11), with their respective x and y first-order Gaussian derivative kernels defined in (4), and is given by

$$M_L(x, y) = \sum_{i=x, y} \nabla g_{\sigma_R}^i(x, y) * M_V^i(x, y) \quad (13)$$

where the Gaussian derivative kernels $\nabla g_{\sigma_R}^i$ are defined in (4) and $M_L(x,y)$ gives large positive values at ridge points and large negative values at valley points. A ridge-enhanced disparity map $M_R(x,y)$ can now be created using

$$M_R(x,y) = M_E(x,y) M_L(x,y) \quad (14)$$

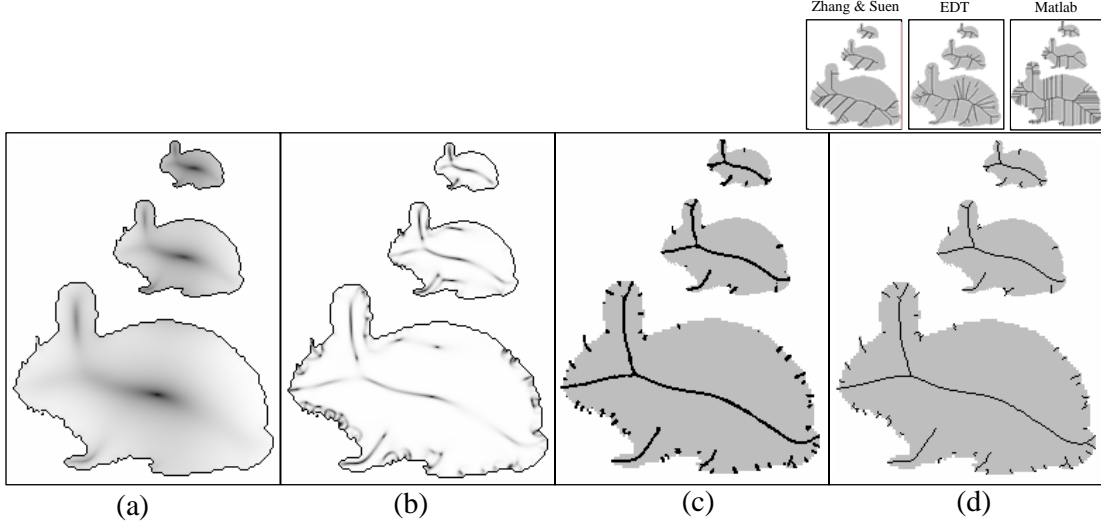


Fig. 8. The process of MGVF skeleton extraction from the disparity map. (a) Disparity map $M(x,y)$. (b) Enhanced disparity map $M_E(x,y)$. (c) Ridge extracted using a local ridge detection algorithm ($N_D=14$). (d) Shape axes ridges thinned to unit-width skeletons. Results from other skeletonization algorithms (see section 5) highlight the comparatively scale-invariant nature of the proposed MGVF-based skeleton.

The local maximas on the ridge-enhanced map $M_R(x,y)$ are detected using the same ridge detection algorithm described in Fig. 6 and the results are shown in Fig. 8c. Disparity-weighted morphological thinning is used to thin the ridge pixels to unit-width skeletons (see Fig. 8d). Skeletal connectivity is maintained while ridge pixels are progressively eroded from outside-in. Removal preference is given to pixels with lower disparity values to ensure the skeletons reside as close to the axes of maximum local vector field directional disparity as discretely possible. This thinning stage is not computationally demanding as the maximum ridge width is no more than three pixels wide. The intentional use of a non homotopy-preserving local ridge detection algorithm is to encourage disconnectivity between structural and textural skeletons. However, connectivity is sought within the structural skeleton. These conflicting requirements pose several challenges to the skeletonization process and will be discussed next.

4.3 Maintaining Structural Skeleton Connectivity

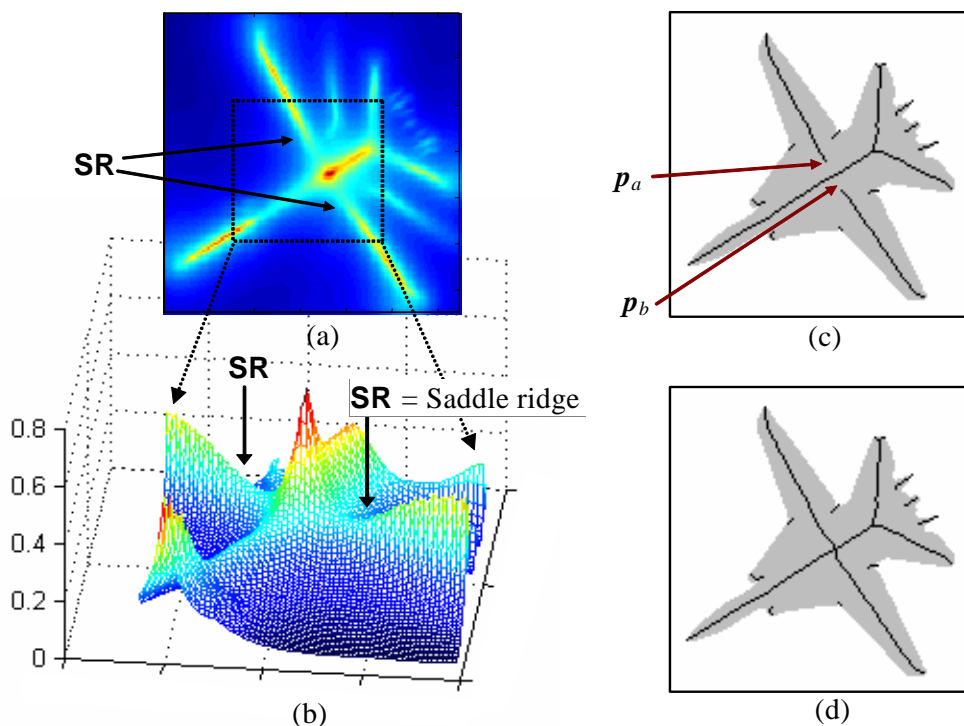


Fig. 9. Broken structural skeleton connectivity. (a) The disparity map $M(x,y)$ and (b) a zoomed-in 3D mesh plot of the disparity surface showing the saddle ridges. (c) Skeletons extracted by using local maximas results in disconnectivity at localities of broad saddle ridges (**SR**). (d) The connected skeleton after applying the DUC to end points p_a and p_b .

Simple local maxima analysis cannot maintain connectivity across saddle ridges such as those shown in Fig. 9. The cross-shaped skeleton of the plane (F4 Tomcat) forms the structural skeleton and should be connected. In order to connect such skeletal segments to an existing structural skeleton, a modified version of the *directional-uphill climbing* (DUC) algorithm proposed by Shih and Pu [26] was adopted. It was originally used in maxima tracking on an Euclidean distance transformed surface but is now applied to the disparity map surface defined by $M(x,y)$ in (9). Its main characteristics are that it operates on a small 3×3 square neighborhood N_p , it employs directional information of the current skeletal end points and it add skeletal pixels to current end point by climbing up the slope with the steepest gradient. The DUC starts at the unconnected textural skeleton end point p that has the lowest disparity value $M(p)$. The algorithm appends a skeletal point to the current end point p by determining which of its eight directional-neighbor has the steepest disparity gradient from p . Once this neighboring point p_{next} has been identified, DUC is then iteratively applied to p_{next} until the current neighborhood N_p

contains a structural skeletal pixel. This process is then repeated for the next textural skeleton end point with the lowest disparity value.

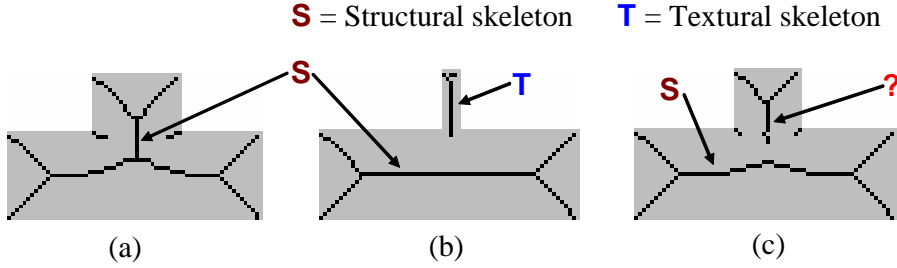


Fig. 10. Is the disconnected skeletal segment structural or textural? Shapes in (a) and (b) have obvious structural and textural skeletons. (c) This case is not so clear cut.

The connection of broken skeletal segments gives rise to another problem. If disconnected segments were to be reconnected, how would one determine if the skeletal segment in question should be part of the structural skeleton or remain an unconnected textural skeleton? This distinction is important since applying DUC to all skeleton end points would result in a single structural skeleton with numerous insignificant medial branches. Some criterion should be used to determine if DUC should be applied to a skeletal end point. Using a region width criterion to measure the structural saliency of a shape, it is easy to differentiate between structural and textural skeletons in the cases shown in Fig. 10a and 10b. However, the protruding flange in Fig. 10c looks prominent enough to be part of the overall shape structure but its skeleton remains disconnected from the structural skeleton. Should connectivity be applied? A single parameter criterion is introduced to decide if a given end point should be connected to the structural skeleton or it should remain unconnected. Fig. 11 shows an object that is defined by a set of 2D boundary points $B = \{\mathbf{b}_i\}_{i=1}^N$ and there is a set of 2D skeletal end points $P = \{\mathbf{p}_j\}_{j=1}^M$ detected within the interior of the object. To determine if DUC should be applied to end point \mathbf{p}_j , a circle c_j with radius r_j is first obtained, where r_j is given by

$$r_j = \lambda R_j, \quad \text{where} \quad R_j = \min_{1 \leq i \leq N} \left\{ \left\| \mathbf{b}_i - \mathbf{p}_j \right\| \right\} \quad (15)$$

and $\| \cdot \|$ gives the Euclidean distance between two points. The circle c_j is termed the λ -maximal disk of \mathbf{p}_j since λ relates its radius r_j to R_j , the radius of the maximal disk centered about \mathbf{p}_j . The parameter $\lambda \in (0, 1]$ is called the *connectivity factor* and is used to control the size of the λ -maximal disk. DUC is only applied to end point \mathbf{p}_j if there is at

least one structural skeletal pixel that lies within the λ -maximal disk region. As such, the connectivity factor λ has a direct influence over the sensitivity of the process that recruits disjointed skeletal segments to form part of the structural skeleton.

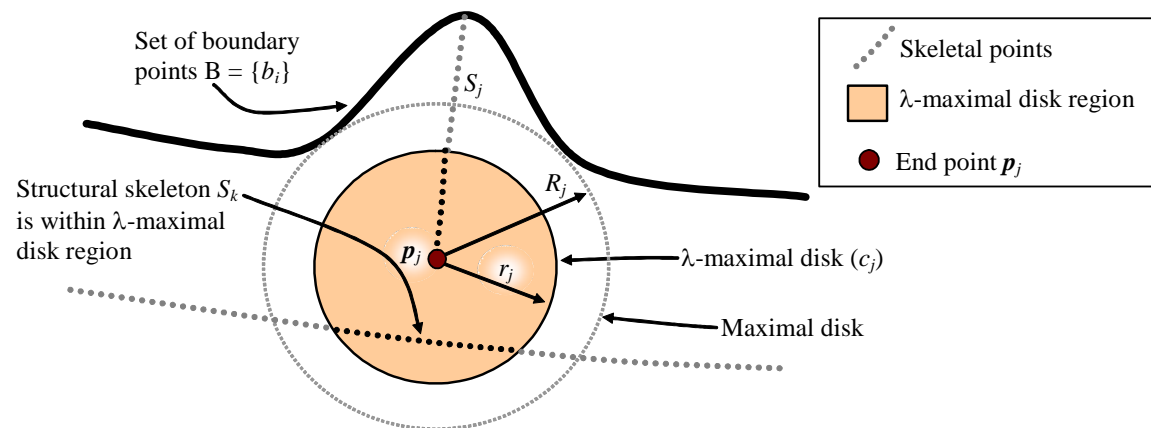


Fig. 11. The λ -maximal disk region (shaded) of connectivity centered about the end point p_j . DUC is applied at p_j since structural skeletal segment S_k is within this region.

4.4 Spurious Textural Skeletons

As can be observed in Fig. 12a, textural skeletal segments can be seen emanating from concave boundary points. Points on these segments do not represent center of maximal disks and should not be classified as skeletons in the traditional sense [6]. We term these *spurious textural skeletons*. They arise because ridges in the disparity map $M(x,y)$ are not confined to areas within the shape but also in its background region (see Fig 12b). These ridges inadvertently ‘leaks’ into the shape’s interior. In the process of extracting ridges within the shape, these spurious segments are also extracted. They can be removed with further post-processing that involves connectivity analysis of ridges within and without the shape. However, in our subsequent shape analysis applications, we found these short spurious skeletal segments to be relatively insignificant and can be left as they are.

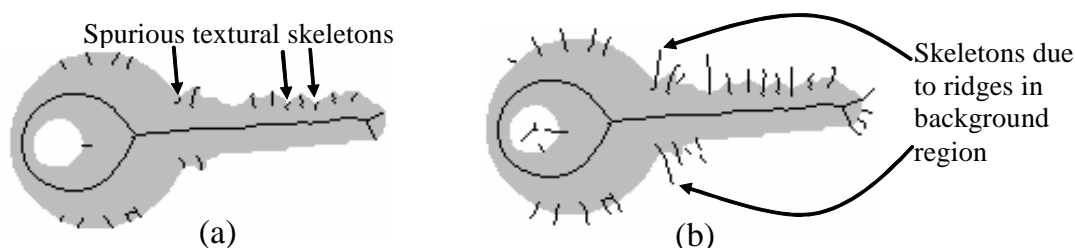


Fig. 12. (a) Presence of spurious textural skeletons at concave boundary areas. (b) Skeletal segments extract within the shape and its background.

5 Experimental Results and Discussion

Several experiments are presented to highlight the characteristics of the MGVF skeleton. Comparative results from two other skeletonization algorithms (one using an iterative approach and the other a non-iterative approach) are also presented. The popular iterative two-pass parallel thinning algorithm by Zhang and Suen [32] was implemented with appropriate improvements suggested by Lu and Wang [15] incorporated. This will be called the *Zhang & Suen* algorithm. For the non-iterative algorithm, the skeleton is extracted from the distance transform of the binary image given by

$$D(\mathbf{p}) = \min\{dist_E(\mathbf{p}, \mathbf{q}), \mathbf{p} \in O, \mathbf{q} \in \bar{O}\} \quad (16)$$

where \mathbf{p} are all pixels within the object O and \mathbf{q} are background pixels defined as \bar{O} . The distance between \mathbf{p} and \mathbf{q} is determined by the squared Euclidean distance metric given by

$$dist_E(\mathbf{p}, \mathbf{q}) = (p_x - q_x)^2 + (p_y - q_y)^2 \quad (17)$$

To reduce computational requirements, the distance transform computation was confined to pixels within the binary shape region. The local ridges in the distance-transformed surface given in (16) are extracted and thinned in the same manner as that described for the proposed MGVF algorithm. The end point connectivity analysis described in section 4.3 is similarly adopted to connect broken ridges extracted from the Euclidean distance transformed image with parameter settings similar to those used in the MGVF algorithm. This algorithm will be called the *EDT* algorithm. In several experiments, skeletonization results obtained from the binary morphological function *bwmorph()* provided by the Matlab® Image Processing Toolbox [16] were also presented to highlight the limitations of the native skeletonization function provided by Matlab® version 5.3. Unless indicated otherwise, the parameters adopted by the MGVF algorithm are as follows: $\sigma_H=1.0$, $\sigma_D=1.5$, $\sigma_V=1.0$, $\sigma_E=1.0$, $\sigma_R=1.0$, $\alpha=0.9$ and $\lambda=0.7$.

5.1 Effects of Varying Smoothing Parameter α

The smoothing parameter α in (5) has a significant influence on the characteristics of the extracted skeleton. With a small value of $\alpha=0.5$, the MGVF generated contains a higher proportion of lower spatial frequency components. As a result, high frequency boundary artifacts such as the serrations on the key shape in Fig. 13a are de-emphasised

(i.e. textural skeletons are fewer and shorter). As α is made larger (e.g. $\alpha=0.9$), even minor serrations are now extracted. Fig. 13b shows the medial axes extracted from the Euclidean Distance Transform (EDT) image of the key shape. The extracted EDT medial axis is a good approximation to a subset of the medial locus defined by Blum's medial axes [6], where the locus is defined by the center of all maximal disks in the interior of the shape that contact the boundary in two or more separate positions. As observed in Fig. 13a, when $\alpha=0.50$ and $\alpha=0.90$, the MGVF structural skeletons do not conform to the medial locus defined by Blum. This is because the summation of the different spatial resolution levels employed in generation of the MGVF results in a non-trivial combination of spatial frequencies arising from the smaller inner boundary (hole in the key) and the larger outer boundary. Conformity to the Blum's medial locus improves as $\alpha \rightarrow 1$, as can be observed when $\alpha=0.99$ but so does the sensitivity to boundary artifacts.

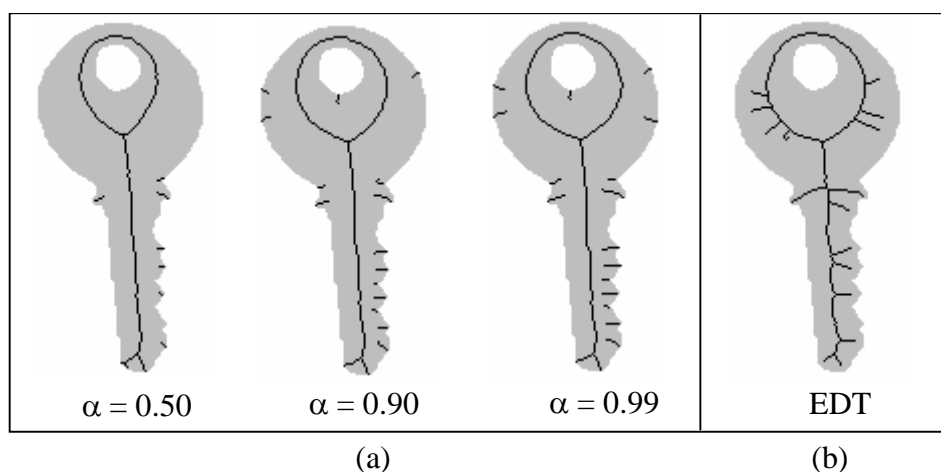


Fig. 13. Skeletons extracted from a key shape using (a) MGVF algorithm with $\alpha = (0.50, 0.90, 0.99)$. (b) Skeleton extracted by the EDT algorithm.

The α -dependent variability of the structural skeleton's position is less significant in shapes that have no holes since no inner-outer boundary interaction exist. However, if these shapes are sufficiently narrow, like the sine-modulated elongated bars depicted in Fig. 14, the structural skeleton can still be significantly influenced by the frequency characteristics of the boundary contour. Notice how in Fig. 14a, the structural skeleton in the interior of the shape begins to undulate along with the boundary modulation when a high value of $\alpha=0.9$ was used to preserve more of the boundary's high frequency components. Alternatively, low values of $\alpha=0.5$ allow these structural skeletons to 'straighten out'. Morse et al. [18] presented similar experiments to show how the

boundary modulation amplitude and the shape's width influenced the 'straightness' of the cores extracted. In the case of the MGVF-based skeleton, the smoothing parameter α can be employed to regulate the influence of boundary perturbations on the structural skeleton. The prominence of the textural skeletons extracted at the boundary depends on both the frequency of the boundary modulation as well as the value of α . It is possible to extract no textural skeleton if both the frequency of the boundary modulation and α are sufficiently low (see Fig 14a when $\alpha = 0.5$) or significant ones when both the boundary modulation frequency and α were higher (see Fig 14b when $\alpha = 0.9$). As observed in Fig. 14b, a lower value of α can also be used to increase the disconnectedness between the structural and textural skeleton. In summary, the experiments presented in this section have highlighted the various influences the smoothing parameter α has on the characteristics of the extracted structural and textural skeletons.

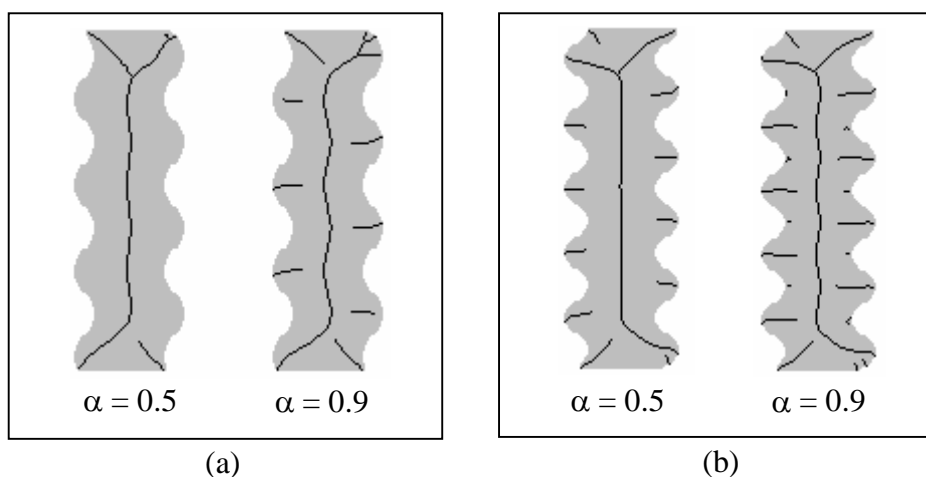


Fig. 14. Skeletons extracted from sine-modulated bars using the MGVF algorithm with $\alpha = (0.5 \text{ and } 0.9)$ and $\lambda = 0$, at sine modulation frequencies of (a) f and (b) $1.3f$.

5.1 Scale Invariant Skeletons

Extracting scale invariant skeleton is particularly important when the shape cannot be easily scale-normalized, such as when partial occlusion prevents the estimation of the actual area of the object. It is therefore desirable that the extracted skeleton should not differ significantly over large scale variations. Fig. 15a shows that the MGVF algorithm is able to extract consistent structural skeletons with scale changes spanning two octaves. Textural skeletons, on the other hand, became more numerous with increasing scale. This was due to the increasing prominence of the boundary discretization noise resulting from

the binary zooming algorithm used. This is not necessarily a shortcoming of the MGVF algorithm as the textural skeletons rightfully represent the increasingly visible boundary texture. Such discrepancy does not produce problems for scale invariant analysis since the structural skeleton remains disconnected from the textural skeletons. This however, cannot be said of the skeletons produced by the Zhang & Suen (see Fig. 15b) and EDT (see Fig. 15c) algorithms. As the scale was increased, more runners were observed emanating from the boundary to the main skeletons in the interior. In these cases, scale invariant shape analysis would require some additional post-processing such as insignificant branch pruning.

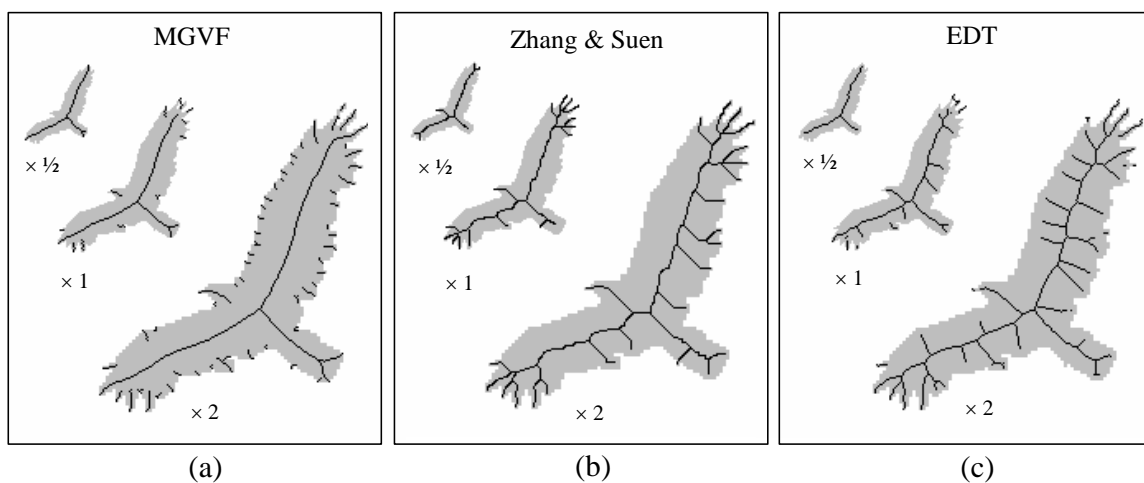


Fig. 15. Skeletons extracted at the indicated scales using the (a) MGVF ($\alpha = 0.8$, $\lambda = 0.7$), (b) Zhang and Suen and (c) EDT ($\lambda = 0.7$) algorithms. Images are 257×257 pixels.

5.2 Rotation-Invariant Skeletons

The rotation invariance of the MGVF algorithm is clearly observed in Fig. 16a since only isotropic Gaussian-based operators were employed in generating the MGVF and disparity maps. Good isotropic behavior is difficult to achieve in iterative algorithms such as those reviewed in [13]. In sequential algorithms, the resulting skeleton depends on the order in which the pixels are processed. In the case of parallel algorithms that remove border points in each sub-iteration, the result is dependent on the order of the sub-iteration [13]. Parallel thinning algorithm such as the MB2 algorithm [5] claims that rotation invariance can only be guarantee for angles multiples of $\pi/2$ radians because they are 4-isotropic, this means the thinning action is exactly the same in four cardinal

directions (N, E, S, W). Such limitations, are shared by Zhang & Suen's iterative parallel thinning algorithm (see Fig. 16b). Notice that the skeleton extracted from the shape that was rotated by an odd 30° is very different from that rotated by $\pi/4$ radians (45°). Skeletons extracted from distance transforms are only rotationally invariant if the distance metric employed is truly Euclidean. Unfortunately, the more computationally efficient city block and chessboard distances are not invariant to arbitrary rotation. Since the EDT algorithm uses the squared Euclidean distance metric given in (17), complete rotation invariance is observed in Fig. 16c. The least consistent result seen in Fig. 16d was produced by the binary morphological *bwmorph()* function provided by the Image Processing Toolbox in Matlab® version 5.3.

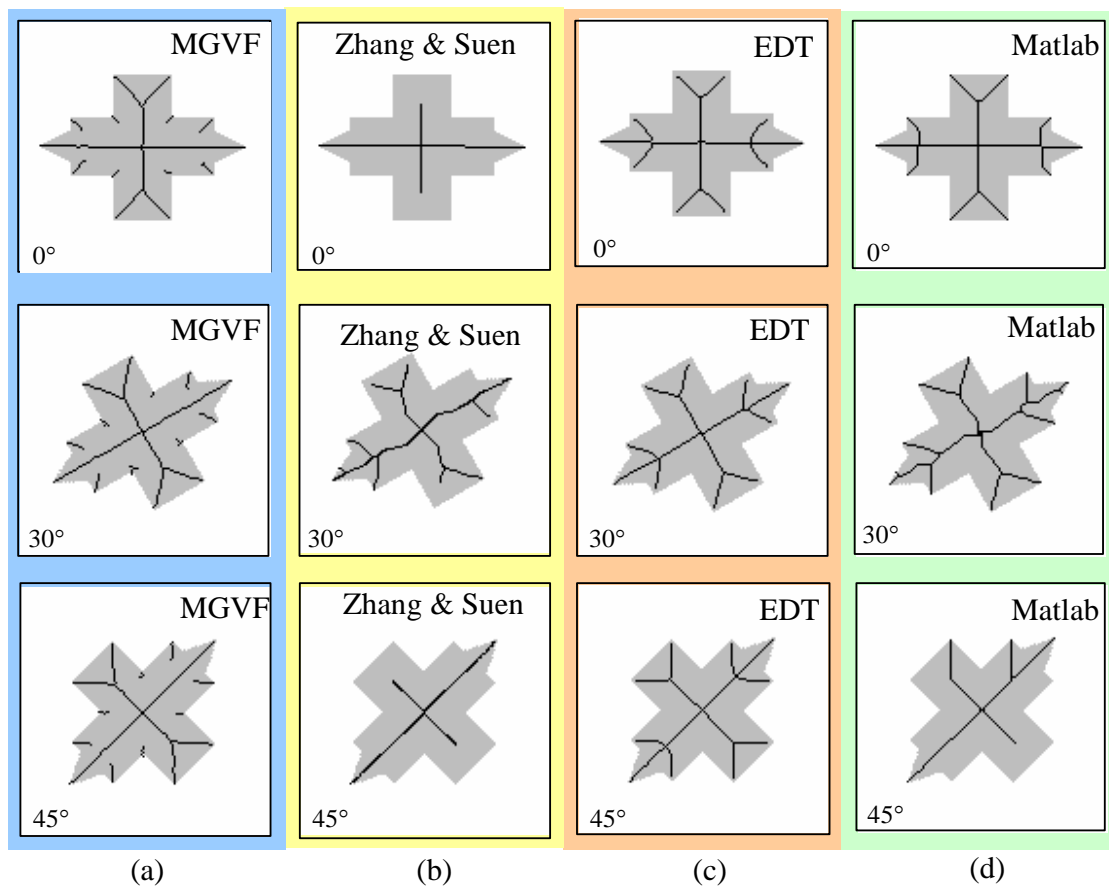


Fig. 16. Results of the rotation invariance experiment. Skeletons extracted from the (a) MGVF, (b) Zhang and Suen, (c) EDT and (d) Matlab® *bwmorph('skel')* algorithms. Rotation angles are indicated at bottom-left corner of each image.

5.4 Analysis of Boundary Noise Sensitivity

Despite the large numbers of skeletonization algorithms developed in the past, there have not been many techniques proposed for quantitative analysis of skeletons. Lam and Suen in [12] proposed three different measures of distance to evaluate the closeness of an extracted skeleton to a reference or ideal skeleton, namely mean-squared distance based on equispaced points, mean-squared projection distance and mean-squared distance based on string matching and dynamic programming. The experimental setup employed here is similar to that in [1] and the distance measure used is similar to the projection distance employed in [12]. The goal of this experiment is to evaluate the accuracy of the skeleton extracted from the MGVF in the presence of boundary noise. A known reference skeleton with N skeletal points is first extracted from an undistorted shape, which in this case is a square. The boundary of the square is then perturbed in the x and y directions by varying degree of zero-mean Gaussian noise with varying standard deviation σ_n . For each test image generated, the Euclidean distance between the each skeletal pixel in the reference skeleton and the closest skeletal pixel of newly extracted skeleton is computed, resulting in a distance array x_i , where $1 \leq i \leq N$. The variance of the skeletal discrepancy [1] is given by

$$\text{var}(X) = E(X^2) - E^2(X) \quad (18)$$

where
$$E(X) = \frac{1}{N-1} \sum_{i=1}^N x_i \quad \text{and} \quad E(X^2) = \frac{1}{N-1} \sum_{i=1}^N x_i^2 \quad (19)$$

$E(X)$ is the mean Euclidean distance and $E(X^2)$ is the mean-squared Euclidean distance. The standard deviation of the skeletal discrepancy, SD is given by

$$SD = \sqrt{\text{var}(X)} \quad (20)$$

Fig. 17 shows the variations in the mean value of SD for the MGVF and the EDT skeletons, for $0 \leq \sigma_n \leq 4$. The experiment was conducted 100 times and for each run, the value of σ_n was incremented by steps of 0.2. The same randomly generated noisy square was used for both the MGVF and EDT algorithms. The results show that the skeletal discrepancy of skeletons extracted using the proposed MGVF algorithm is consistently lower than those extracted using the EDT algorithm, especially at higher levels of the Gaussian boundary noise.

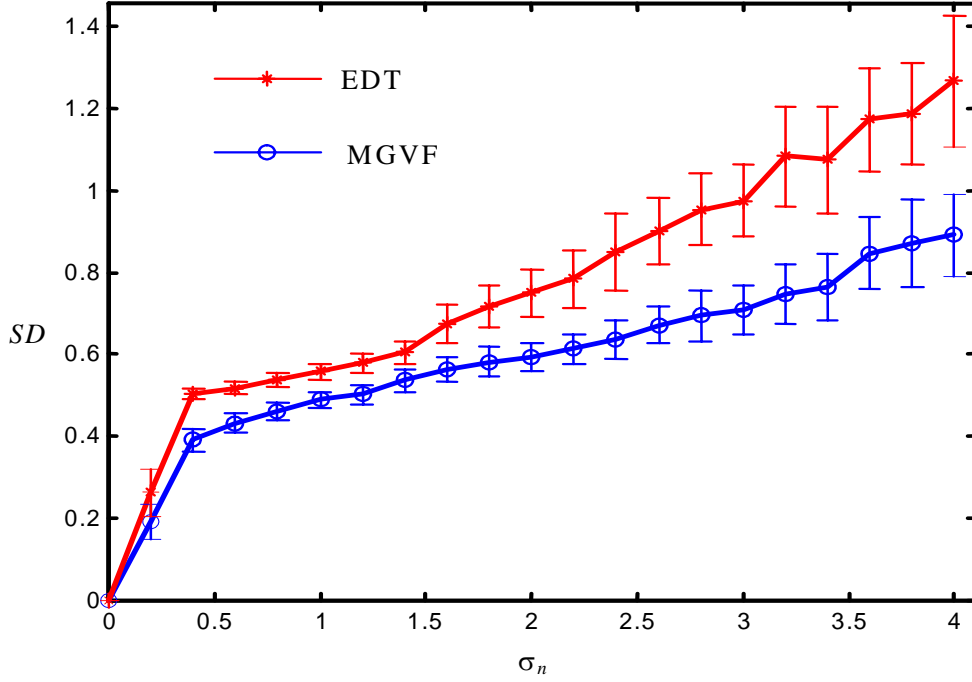


Fig. 17. The plots of standard deviation (SD) in the skeletal discrepancies of the extracted MGVF and EDT skeletons for varying degrees of boundary noise. The parameter settings for the MGVF and EDT algorithms were $(\alpha=0.8, \lambda=0.7)$ and $(\lambda=0.7)$, respectively. Error bars drawn at each σ_n value is the standard deviation obtained after 100 runs and they are plotted about their respective mean values.

The boundary noise robustness of the skeletons extracted using the MGVF algorithm is due to the fact that the gradient vector field within the interior of the shape is derived mainly from the low spatial frequency components of the shape's boundary. The skeletal information embedded with the gradient vector field disparity is therefore less susceptible to high frequency distortions present in the noisy and jagged boundaries. This can be clearly observed from the relatively consistent structural skeletons extracted using the MGVF algorithm, even at noise levels of $\sigma_n=4.0$, as shown in Fig. 18. The Euclidean distance transform on the other hand, faithfully transmit the boundary perturbations into the interior region (see EDT distance maps in Fig. 18), which inadvertently distorts the skeletons. Moreover, the many extraneous skeletal branches will complicate subsequent skeletal analysis of the noisy shapes. In contrast, the MGVF skeletons have reasonably consistent structural skeletons that represent a general square shape. At the same time, the noisier boundaries are also noted by the increasingly more prominent textural skeletons. Since these two categories of skeletons remain unconnected, subsequent skeletal analysis of the noisy shapes remains relatively straightforward.

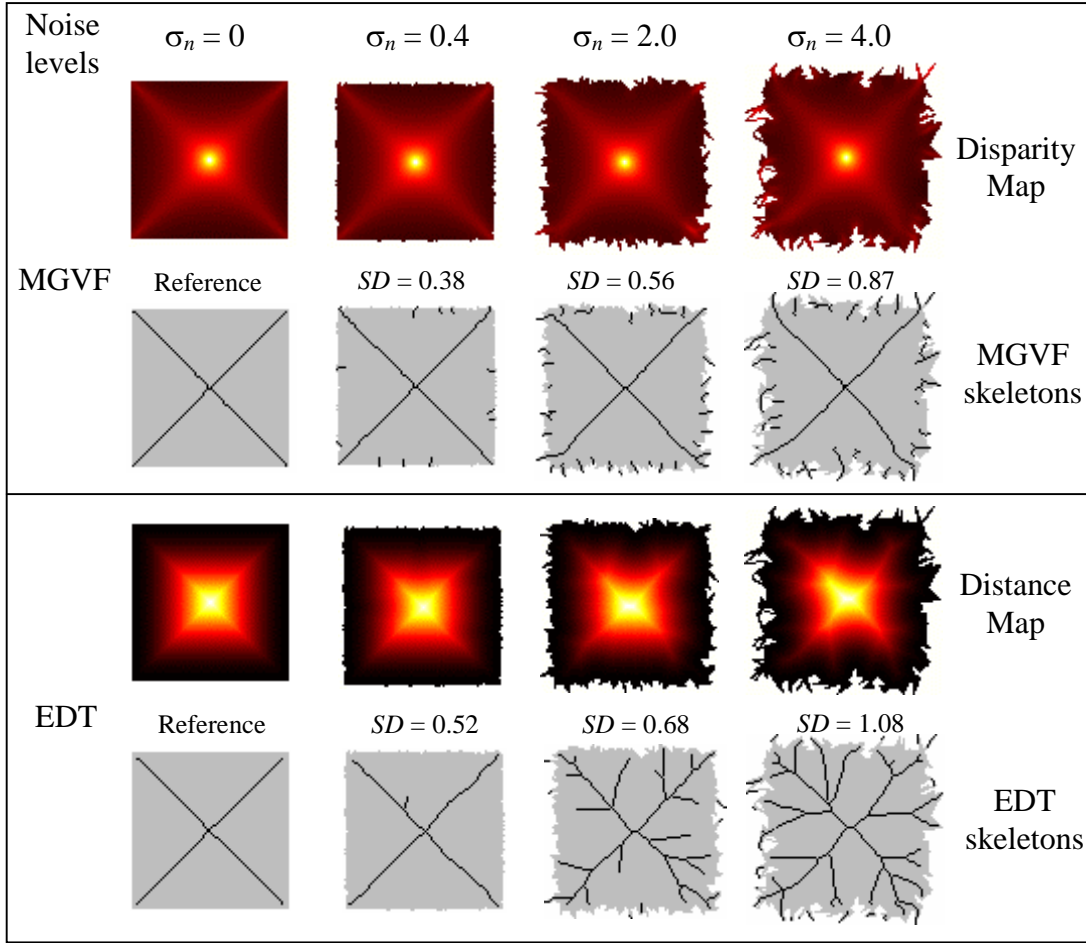


Fig. 18. Skeletons extracted from the disparity and distance maps using the MGVF and EDT algorithms, respectively at selected boundary noise levels of $\sigma_n = 0, 0.4, 2.0$ and 4.0 . The standard deviations SD of each extracted skeleton from their respective reference skeleton are also indicated. The skeletons at $\sigma_n = 0$ are the reference skeletons.

5.5 Robustness to Boundary Texture

One of the strength of the proposed MGVF algorithm is its ability to control the proportion of low and high spatial frequency components used to create the gradient vector field. By choosing a lower value of α in (5), we can reduce the influence of boundary texture on the gradient vector field in the shape's interior. This will facilitate the extraction of relatively consistent structural skeletons that represent the object's gross shape form, as observed in the boundary textured hares example in Fig. 19. What is unique about the MGVF-based skeletonization algorithm is the fact that the textural skeletons near the boundaries are simultaneously extracted together with the structural

skeleton. Both these classes of skeletons remain unconnected, which makes possible the simultaneous skeletal analysis of both the object's gross shape form and its boundary characteristics. This approach is preferable to techniques that employ curvature flow smoothing of the shape boundary [10], [17]. It is not possible to find one level of smoothing that can allow one to analyze both the gross shape and detail boundary texture. If sufficient shape smoothing has been applied such that the general shape class can be visualized, all notion of boundary texture would have been removed.

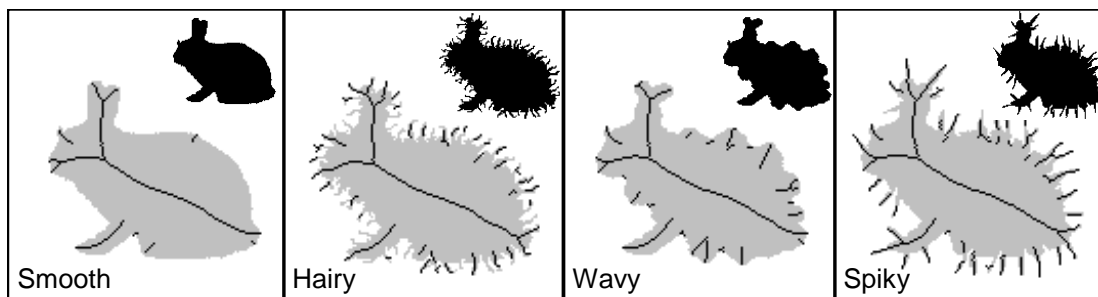


Fig. 19. Skeletons extracted from four different boundary-textured hare shapes using the proposed MGVF algorithm with parameters $\alpha=0.6$ and $\lambda=0.7$. The hairy, wavy and spiky textured images were created from the original smooth hare image.

5.6 Example Application - Character Recognition

In optical character recognition (OCR) applications, textual symbols are first isolated and then thinned to a unit-width skeleton before feature extraction and analysis. This experiment studies the comparative performance of the MGVF algorithm when applied to this task. The test images used are shown in Fig. 20.

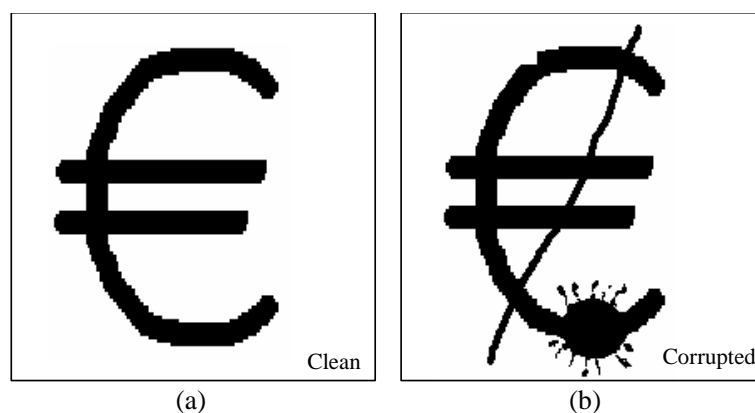


Fig. 20. Test images for OCR experiment. (a) Euro symbol with boundary discretization noise and (b) one corrupted with a line scratch, ink splotch and partially occlusion.

Fig. 21b shows the skeleton extracted from the uncorrupted euro symbol using the EDT algorithm. Notice how boundary discretization noise can sometimes result in spurious medial branches. These boundary discretization artifacts also result in a slightly jagged skeleton when the Zhang and Suen [32] thinning algorithm is employed, as can be observed in the enlarged image segment at the bottom-left corner of Fig. 21c. In contrast, the incorporation of a larger proportion of low spatial frequency components in the interior of the shape resulted in a smoother skeleton (see Fig. 21a), much like those obtained by Shroff and Ben-Arie in [27]. Fig. 21d shows the result obtained using the binary morphological function *bwmorph*('skel') from Matlab® [16], which is extremely sensitive to boundary discretization noise.

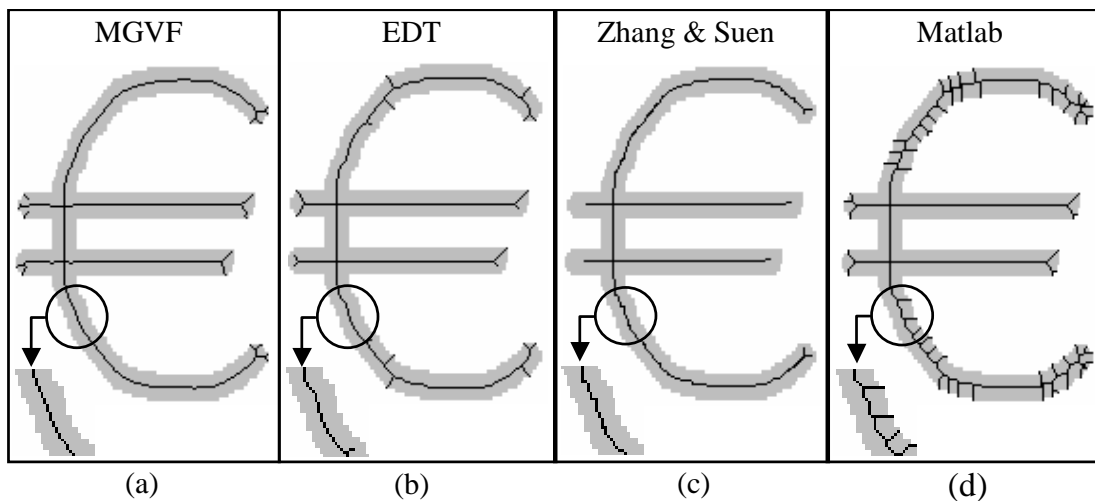


Fig. 21. Extracted skeletons for the clean euro symbol using the (a) MGVF, (b) EDT ($\lambda = 0.7$), (c) Zhang and Suen and (d) Matlab's *bwmorph*() algorithms.

Another useful performance measure is to observe how well each skeletonization algorithm is able to consistently replicate the extracted skeleton when the same symbol is corrupted with various types of boundary artifacts. As expected, the EDT result in Fig. 22b shows extra medial branches where the angular occlusion is present and at some locations where the line scratch intersects the symbol. The ink splotch is particularly problematic to the connectivity-seeking Zhang & Suen thinning algorithm in Fig. 22c. The splatter tails around the splotch provide anchor points that allowed the thinning algorithm to create many spurious branches in responses to the need to maintain skeleton connectivity during the boundary erosion iterations. This same artifact can also be observed along the line scratch. Matlab's *bwmorph*() function suffers from essentially the

same problem as can be seen in Fig. 22d. Despite the various distortions, the MGVF algorithm is still able to extract a smooth and consistent structural skeleton, as shown in Fig. 22a. These results support the case for a medial representation that does not enforce connectedness between medial branches originating at the boundary and those located in the interior of the shape. Minor boundary artifacts, such as the thin line scratch, should generate unconnected textural skeletons that remain close to the boundary. However, the skeletons of the two horizontal segments, which are actually parts of the euro symbol, should be part of the structural skeleton. Since their widths are comparable with the ‘C’-shaped segment, the structural skeleton connectivity algorithm described in section 4.3 (with $\lambda=0.7$) will ensure that these two horizontal skeletons do connect at their respective intersections with the ‘C’-shaped skeleton. Together, they form a connected structural skeleton that correctly represents the original euro symbol.

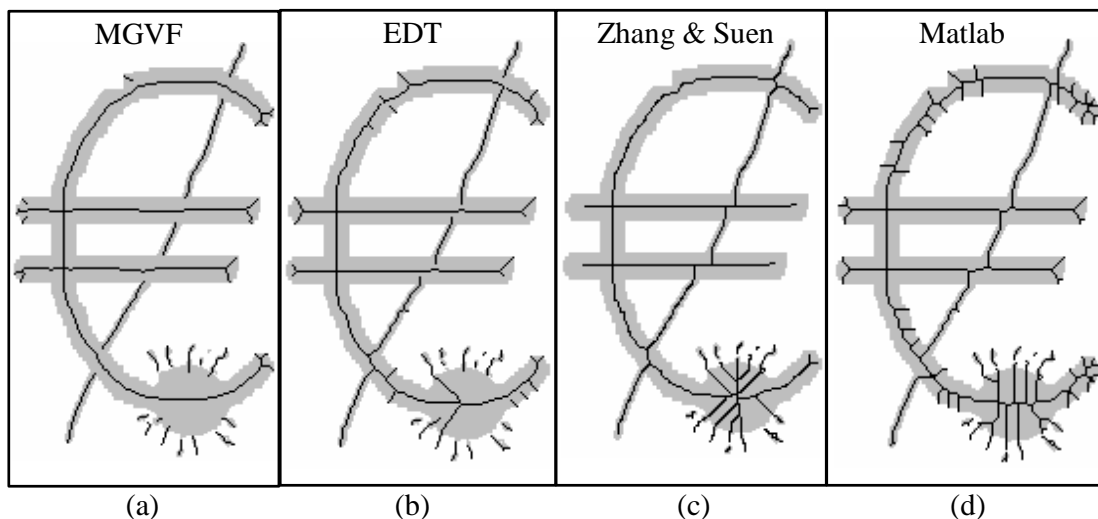


Fig. 22. Extracted skeletons for the corrupted euro symbol using the same four algorithms and parameter settings described in Fig. 21.

5.7 Computational Cost

The computation costs of the four skeletonization algorithms used in the previous sections were compared using their respective Matlab implementations. Even though the compute times recorded do not necessarily indicate the best performance of each algorithm (since they are not optimized), it does give an indication of their relative computational efficiency. The only exception in this case is Matlab’s native binary thinning algorithm *bwmorph()* [16], which has been optimized through compilation. Compute time measurements were obtained for the skeletonization of two different

129×129 pixel-sized images and they are tabulated in Table 1. The two images (Eight and Hare) were appropriately selected to demonstrate how the maximum width of the shape affects the computational times of the algorithm.

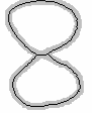

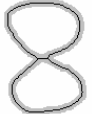





Technique	Time (sec.)	Results	Time (sec.)	Results
MGVF				
Generate vector field	1.04		1.04	
Thin and connect ridges	0.82		1.76	
Total compute time	1.86		2.80	
EDT				
Distance transform	3.84		7.52	
Thin ridges	0.49		1.21	
Total compute time	4.33		8.73	
Zhang and Suen				
Total compute time	0.71		7.91	
Matlab – <i>bwmorph</i>				
Total compute time	0.22		1.15	

Table 1. Computational cost incurred in computing the skeletons of two different images (Eight and Hare) using the MGVF, EDT (without connectivity analysis), Zhang & Suen and Matlab® *bwmorph()* skeletonization algorithms. The times shown were measured on a 1.6 GHz Pentium-4 Personal Computer with 255 MB DRAM, running Windows 98® and Matlab® version 5.3. Matlab codes for building the MGVF’s Gaussian pyramids were based on Simoncelli’s pyramid tools available from [30].

As expected, the fastest computation times for both images were registered by Matlab’s own optimized *bwmorph()* algorithm. It was 8.5 times and 2.4 times faster than the MGVF-based algorithm for the Eight and Hare images respectively. Its speed advantage dropped significantly as the maximum width of the shape increased. This shape-width variability in skeletonization time is generally undesirable but it is an inherent characteristic of morphologically-based thinning algorithms such as *bwmorph()* and Zhang and Suen parallel thinning algorithms, which iteratively thins the given shape from the boundary inwards, much like the peeling of the layers of an onion. This variability is also true of the EDT algorithm as the number of distance transform computation depends on the number of pixels in the shape. Zhang & Suen algorithm exhibited a 11-fold variation in compute time between the two images, compared to the

EDT and MGVF algorithms' 2-fold and 1.5-fold variations respectively. In the case of the MGVF algorithm, the variation is mainly due the skeleton-dependent ridge thinning and connectivity analysis processes. The MGVF algorithm was consistently much faster than the EDT algorithm but may be slower than the Zhang & Suen's parallel thinning algorithm for shapes with narrow widths.

6 Conclusions

It was suggested that the homotopy-preserving approach of many skeletonization algorithms may not produce medial structures suitable for robust and efficient shape representation under noisy conditions. This paper proposes that shapes should be described using structural and textural skeletons. It has been shown that such skeletons can be recovered from the loci of local directional disparity maximas in the proposed multiresolution gradient vector field. The advantages of this dual skeletal representation were highlighted by several comparative experiments, which showed that the MGVF-based structural skeletons can be extracted in a consistent manner despite the presence of boundary artifacts such as noise, texture, line scratches. They are also invariant to large scale changes and arbitrary rotation. Subsequent skeletal analysis of shapes can also be simplified by the fact that these structural skeletons, which describe the general shape of objects, remained largely disconnected from the numerous textural skeletons at the boundary. The computational cost of the MGVF skeletonization algorithm was also shown to be the least shape-width dependent of all the algorithms evaluated. This is generally viewed as advantageous as applications adopting this algorithm will have a more consistent response time when applied to objects of varying sizes. Its estimated computational cost is lower than that of the EDT algorithm. It is also more efficient than the popular Zhang & Suen thinning algorithm for shapes with a broader width.

7 References

- [1] G.J. Abdel-Hamid, Y.H. Yang. Multiresolution skeletonization: an electrostatic field-based approach. Proc. of Int. Conf. on Image Processing 1994 (1) pp. 949-953.
- [2] C. Arcelli, G. Sanniti Di Baja. Ridge points in euclidean distance maps. Pattern Recognition Letters 13 (1992) 237-243.

- [3] J. August, K.S. Siddiqi, S.W. Zucker. Ligature instabilities in the perceptual organization of shape. Proc. of Conf. on Computer Vision and Pattern Recognition 1999 (2), pp. 42-48.
- [4] J. Ben-Arie, Z. Wang. Hierarchical shape description and similarity-invariant recognition using gradient propagation. International Journal of Pattern Recognition and Artificial Intelligence 15 (8) (2001) 1251-1261.
- [5] T.M. Bernard, A. Manzanera. Improved low complexity fully parallel thinning algorithm. Proc. of Tenth Int. Conf. on Image Analysis and Processing 1999, pp. 215-220.
- [6] H. Blum. A transformation for extracting new descriptors of shape, in W. Wathen-Dunn, ed., Models for the Perception of Speech and Visual Form, pp. 363-380, Cambridge, MA: MIT Press, 1967.
- [7] G. Borgefors. Distance transformation in digital images. Computer Vision, Graphics, and Image Processing 34 (1986) 344-371.
- [8] P.J. Burt. The pyramid as a structure for efficient computation. in A. Rosenfeld, ed., Multiresolution Image Processing & Analysis, pp. 6-35, Berlin Heidelberg New York, Springer-Verlag, 1984.
- [9] A.D.J. Cross, E.R. Hancock. Scale Space vector fields for symmetry detection. Image and Vision Computing 17 (1999) 337-345.
- [10] B.B. Kimia, A. Tannenbaum, S.W. Zucker. Shape, shocks and deformation, i: the components of 2-d shape and the reaction-diffusion space. International Journal of Computer Vision 15 (1995) 189-224.
- [11] B.B. Kimia. On the role of medial geometry in human vision, Journal of Physiology 97 (2-3) (2003) 155-190.
- [12] L. Lam, C.Y. Suen. Automatic evaluation of skeleton shapes. Proc. of Eleventh Int. Conf. on Pattern Recognition 1992 (2) pp. 342-345.
- [13] L. Lam, S.W. Lee and C.Y. Suen, Thinning methodologies – a comprehensive survey, IEEE Trans. on Pattern Analysis and Machine Intelligence 14 (9) (1992) 869-885.
- [14] T.L. Liu, D. Geiger. Approximate tree matching and shape similarity. Proc. Int. Conf. on Computer Vision 1999, pp. 456-462.
- [15] H.E. Lu, S.P. Wang. A comment on ‘a fast parallel algorithm for thinning digital patterns’. Communications of the ACM 29 (3) (1984) 239-242.
- [16] Matlab User Guide – Image Processing Toolbox (version 2), *The MathsWork Inc.*, 1999.
- [17] F. Mokhtarian, A.K. Mackworth A theory of multiscale, curvature-based shape representation for planar curves. IEEE Trans. on Pattern Analysis and Machine Intelligence 14 (8) (1995) 57-62.
- [18] B.S. Morse, S.M. Pizer, D.T. Puff, C. Gu. Zoom-invariant vision of figural shape: effects on cores of image disturbances. Computer Vision and Image Understanding 69 (1) (1998) 72-86.
- [19] R.L. Ogniewicz, O. Kübler. Hierarchic voronoi skeletons. Pattern Recognition 28 (3) (1995) 342-359.

- [20] S.M. Pizer, D. Eberly, D.S. Fritsch, B.S. Morse. Zoom-invariant vision of figural shape: the mathematics of cores. *Computer Vision and Image Understanding* 69 (1) (1998) 55-71.
- [21] S.M. Pizer, K. Siddiqi, G. Székely, J. N. Damon, S.W. Zucker. Multiscale medial loci and their properties. *International Journal of Computer Vision* 5 (2-3) (2003) 155-179.
- [22] H. Rom, G. Medioni. Hierarchical decomposition and axial shape description. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 15 (10) (1993) 973-981.
- [23] T.B. Sebastian, P.N. Klein, B.B. Kimia. Recognition of shapes by editing shock graphs. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 26 (5) (2004) 550-571.
- [24] J. Shah, Gray skeletons and segmentation of shapes, *Computer Vision and Image Understanding* 99(1) (2005), 96-109.
- [25] D. Shaked, A.M. Bruckstein. Pruning medial axes. *Computer Vision and Image Understanding* 69 (2) (1998) 156-169.
- [26] F.Y. Shih, C.C. Pu. A skeletonization algorithm by maxim tracking on euclidean distance transform. *Pattern Recognition* 28 (3) (1995) 331-341.
- [27] H. Shroff, J. Ben-Arie. Finding shape axes using magnetic field. *IEEE Trans. on Image Processing* 8 (10) (1999) 1388-1394.
- [28] K.S. Siddiqi, A. Shokoufandeh, S. Dickinson, S. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision* 35 (1) (1999) 13-32,.
- [29] K.S. Siddiqi, S. Bouix, A. Tannenbaum, S. W. Zucker. Hamilton-Jacobi skeletons. *International Journal of Computer Vision* 48 (3) (2002) 215-231.
- [30] E. Simoncelli. Steerable pyramid toolbox, available from Department of Computer and Information Science, Penn Engineering, University of Pennsylvania at <http://www.cis.upenn.edu/~eero/steerpyr.html>, 2003.
- [31] S. Tari, J. Shah, Local symmetries of shapes in arbitrary dimension. *Proc. of Sixth Int. Conf. on Computer Vision*, 1998, pp.1123-1128.
- [32] T.Y. Zhang, C.Y. Suen. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM* 27 (3) (1984). 236-239.