Star Fields: Improvements in Shape-Based Image Retrieval

Alberto Chávez-Aragón[1], Oleg Starostenko[2] and Leticia Flores Pulido[1]

[1] Universidad Autónoma de Tlaxcala, Calzada Apizaquito s/n km. 1.5, Apizaco Tlaxcala, México
[2] Universidad de las Américas-Puebla, Sta. Catarina Mártir Cholula, Puebla, México

Abstract. Determining the similarity of two images is a very difficult task in both machine and human vision systems. Not mention the semantic gap. Thus, in order to reduce this problem this paper developed a set of methods for retrieving images based on one low level image feature such as shape. We focused on this important feature of the objects because there is evidence that natural objects are primarily recognized by their shapes. In this paper, we proposed an alternative representation of shapes, that we have called two segment turning function (2STF) which has a set of invariant features such as invariant to rotation, scaling and translation. Then, based on 2STF, we proposed a complete new strategy for computing a similarity among shapes. This new technique was called Star Field (SF). The proposed technique, which is made up of a set of new methods, was implemented in a test-bed CBIR system that we called IRONS. IRONS stands for "Image Retrieval based ON Shape".

1 Introduction

Today huge amounts of new digital documents are available around the world. Every day different types of digital documents such as text, image, video, audio, and animation, among others, are added to the Internet or similar technologies. However, most current search engines' algorithms use text as a principal document descriptor. Techniques which use different descriptors like shape, color, sound, etc. lag behind text-based techniques. This is why there is a growing need for efficient visual information retrieval algorithms which go beyond the text-based retrieval approach. In other words, there is a lack of reliable and efficient systems to get relevant information contained in multimedia documents. This paper addresses the problem of retrieving documents that contain visual information. Although it is true that content-based image retrieval systems already exist, many of these systems some times retrieve irrelevant documents or documents unrelated to the user's query. This problem is caused by the use of low-level image descriptors; furthermore, these descriptors hardly have a semantic weight. Specifically, this work addresses the image retrieval problem based on shape, since shape has a meaning by itself.

© S. Torres, I. López, H. Calvo. (Eds.)
Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 79-90

Received 23/02/07
Accepted 08/04/07
Final version 23/04/07

2 Visual Information Retrieval

The visual information retrieval problem is an extension of the IR problem to the images domain. Visual Information Retrieval can be define as: *"The use of technology to obtain result images from a query based on its visual information"* [13]. In other words, its purpose is to retrieve from a database images or image sequences which are relevant to a query. The Visual Information Retrieval area (VIR) challenge is to go beyond the text search, which describes the images in order to store and recover visual information from digital repositories.

There are several reports about recent researches in the visual information retrieval area. Consequently, many VIR systems have been produced. According to Venters [2] all of these systems can be classified into two main groups: Commercial Image Retrieval Systems and Prototype Research Systems. Among commercial image retrieval applications, the following systems stand out: Excalibur Visual RetrievalWare [6], ImageFinder, IMatch [26], QBIC [22] and Virage [10]. On the other hand, AMORE [3], Photobook [16], PictoSeek [25], SQUID [23], VisualSEEK [11], Black Box [13] and Keyblock [19] are examples of Prototype Research Systems.

3 Shape-Based Retrieval

Perhaps the most obvious requirement of users for VIR systems is to retrieve images by shape, since there is evidence that natural objects are primarily recognized by their shape [9]. Features vectors which represent object shapes contained in images are computed in order to be indexed in a database. The query process works in the same way that color-based and texture-based retrieval work in the sense that a query can be an image. But, unlike color and texture retrieval, shape-based retrieval has another particular way to feed the query into the system. This is by means of sketching. Systems which support this kind of queries must provide the user with a sketch tool [12],[27].

3.1 Shape Representation and Matching

There is no universal definition of what shape is, but it is possible to mention some well accepted definitions. Shape is the outward form of an object defined by its outline; shape is the external appearance of something [7]. In this paper we consider shape as a geometrical pattern, consisting of a set of points, curves, surfaces, solids, etc. Shape matching is considered one of the most difficult aspects of content-based image retrieval since the representation of shapes is often more complex than color and texture. The difficulty lies in the fact that a common shape needs a lot of parameters to be represented explicitly.

4 Shape Representation

Traditionally, a shape is described as a closed polygon. However, the polygonal representation of shapes is not a convenient way for computing the similarity among them. In order to overcome this problem we propose a different representation that we have called two-segment turning function. Our technique is based on tangent space representation but it has some advantages that are outlined below.

4.1 Polygonal Representation

Our strategy for computing similarity among shapes starts out getting the outline of the shape from an image. Basically, we assume as a premise that each image we are working with represents just one object. Besides, the object has been previously separated from the background. That means that our images are binary ones and the objects are represented by white pixels and the backgrounds by black pixels.

The method for getting the outline consists of two main stages. The first step designates one pixel of the object border as the starting point. We choose as starting point the first white pixel which is on the first row that belongs to the object. The second step consists in tracking down those pixels that make up the object border. The tracking task is make in the clockwise direction. Figure 1 shows a result of outline detector algorithm.

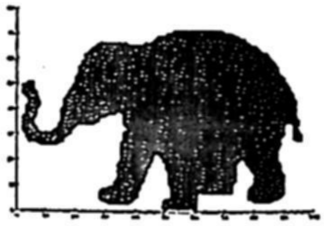

Fig. 1. The image on the left is a binary image and it was the input of the outline detector algorithm The image on the right shows the result given by the algorithm.

Up to now, a closed polygon which represents the object we are interested in is obtained; however, this polygon has plenty of vertices. The next natural step is to reduce the number of vertices so that we can apply an efficient similarity strategy.

4.2 Relevance easure

In order to decrease the number of vertices of a shape it is necessary to calculate what is the relevance of each vertex. The relevance measure K that we use is based on two parameters, the length and the turn angle of two consecutive line segments which share the vertex we want to compute its relevance. The relevance is defined as it is shown in equation 1.

$$K(S_1, S_2) = \frac{\beta(S_1, S_2)l(S_1)l(S_2)}{l(S_1) + l(S_2)} \quad (1)$$

where $\beta(S_1, S_2)l(S_1)$ is the turn angle at the common vertex of the segments S_1, S_2, and l is the length function normalized with respect to the total length of the polygonal curve C. The lower value of $K(S_1, S_2)$ is, the less contribution to the shape of the curve of arc $S_1 \cup S_2$ is. To stop the evolution process it is necessary to use a parameter that defines the number of iterations or to use a threshold which represents the permitted range of values for any simplified shape vertex. A curve evolution algorithm makes the former task. Figure 2 shows the results obtained after applying the curve evolution algorithm to a polygon. It is clear that curve evolution algorithm keeps the main visual parts of the original polygonal curve and obviously the amount of information has decreased drastically.

Fig. 2. The closed polygon on the left is the evolution of the original polygon, it has 30 segments. On the other hand, the figure on the right shows a stage of evolution with just 20 segments; in spite of this the main visual parts are maintained.

5 Two-segment Turning Function

The polygonal representation of a shape is not a convenient form to calculate how similar is that shape to another. In order to overcome this problem and make easier and more effective the matching process, we propose a alternative representation that we have called two-segment turning function or $2STF$. Using $2STF$ a polygonal curve P is represented by the graph of a step function, the steps on $x-axis$ represents the normalized arc length of each segment in P, and the $y-axis$ represents the turn angle between two consecutive segments in P. The former feature gives the name to our proposed technique. Figure 3 shows the angle that is taking into account in order to build the $2STF$. The angle is defined by S_2 and the imaginary line that pass through the segment S_1. This form for measuring the angle has an intuitive reason and this is that the angle measures the deviation of the second segment in respect to the first segment direction. It is clear that the angle values are in the interval $[-\pi, \pi]$.

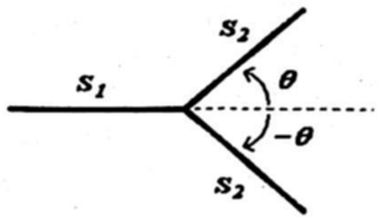

Fig. 3. This figure shows the angle used for $2STF$. The angle is defined by the imaginary line passing through the first segment and the second one. A left turn makes the angle positive and a turn in the clockwise direction makes the angle negative.

6 Star Field

Star Field (SF) is an alternative representation for shapes that allows us to apply a different algorithm in order to obtain a similarity value of two curves. This new algorithm we will propose below does not provide a way to determine the best correspondence among two functions but a very good solution. As a result, Star Field along with a new similarity algorithm are expected to give an easier and faster matching process. A Star Field formally is a torus T_1 x T_2, where T_1 is a circle of length one that represents the length of a polygonal curve and T_2 is a circle that represents the turning direction of digital steps from $2STF$. Nevertheless, most of the time we consider a SF as a window that shows a $2D$ projection of a previously processed torus. This window is made up of stars or points, that is where the name comes from, and each of them represents the relevance measure of each $2STF$ step.

6.1 From $2STF$ to SF

One of the mayor difference between the use of $2STF$'s similarity measure an the one using SF is regarding to the grade of evolution of the digital curves they work with. A star field diagram is basically a $2D$ plane, it is divided horizontally into two section. The upper section holds the stars that represents vertices of concave arcs. On the other hand, lower part holds vertices of convex arcs. Each star on SF is defined by means of two coordinates. The $y-coordinate$ represents the angle between two consecutive segments. Due to the use of $2STF$ for representing a shape, the interval of the turning angle is $[-\pi, \pi]$ radians. However, in the Star Field the angle is normalized in the interval $[0, 1]$. With respect to the $x - coordinate$, these values correspond to the accumulative length of the steps in $2STF$ from the starting point to the current point. In other words, the $x-coordinate$ represent how far is each vertex from the starting vertex and also this distance is normalized.

To illustrate the way a Star Field looks like, imagine that the $2STF$ has just decreasing steps, the Star Field representation of this function will be crowded in the lower part. This kind of Star Fields represents mainly convex shapes. In the same way, if the $2STF$ shows raising steps, that means that it represents a mainly concave figure and the Star Field is crowded in the upper part. Finally, if a step has an angle equal to zero with respect to the previous one, the $y - coordinate$ of the corresponding star has the value .5 in the Star Field, this is because the values of the Star Field go from [0,1] in both directions. Likewise, if two consecutive segments have $-\pi$ or π radians the $y - coordinate$ of the corresponding point in the Star Field has 0 or 1 respectively. To illustrate this, consider figure 4

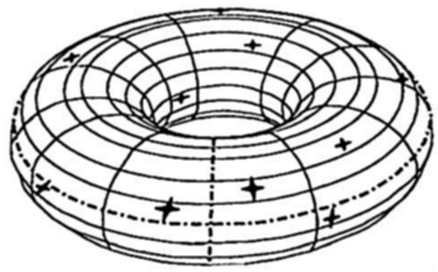

Fig. 4. Star Field, real representation. Actually, Star Fields can be seen as a bending surface like it is shown in this figure. Each star or point, in the Star Field represents the vertex that is shared by two consecutive steps from the equivalent $2STF$.

As we have mentioned before, a Star Field diagram is basically a $2D$ plane. In order to transform a torus into a 2D plane, we imaginatively cut the torus on two places following dotted-lines as it is shown in figure 5. Then, it is necessary to bend the surface, in the sense the arrows show, to get our desired 2D plane. As a result, we obtained a plane similar to the one shown in figure 6

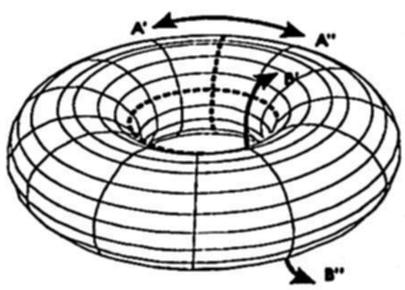

Fig. 5. This figure shows where to cut the torus and in what direction we have to bend it, so that a 2D Star Field representation is obtained.

Since Star Field is based on $2STF$, it has the same invariant characteristics as $2STF$, demonstration of those features are beyond the scope of this paper, for further details see [4].

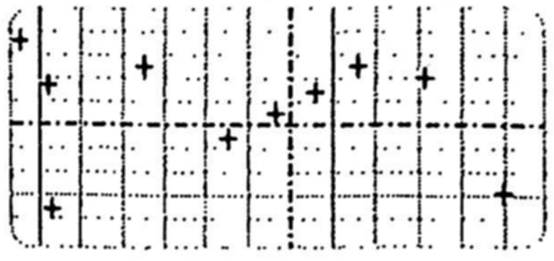

Fig. 6. 2D Star Field representation obtained by means of cutting and bending the torus in the way we explained above.

7 Matching Graph

So far, a new convenient way for representing a polygonal curve has been presented. This new representation give us an idea of who similar two polygonal curves are. However, we need a precise measure. So, we proposed a new similarity measure that makes use of a graph that has particular features. In this section the construction process of this graph is presented.

Given two polygonal curves P_1 and P_2 and their Star Field representations SF_1 and SF_2, the graph G that allow us to compute their similarity is defined as follows. $G = (V, E)$ where V and E are disjoint finite sets. We call V the vertex set and E the edge set of G. Our particular graph G has a set V which consists of two smaller subset of vertices v_1 and v_2. $V = v_1 \cup v_2$, where v_1 is the set of point of SF_1 and v_2 is the set of points of SF_2. On the other hand, E is the set of pairs (r, s), where $r \in v_1$ and $s \in v_2$.

According to previous definition the edges of our graph, that we will call from now on matching graph or MG, consists of two points and each point comes from a different Star Field representation. But also a new restriction will must be introduced, this is stated as follows. $\forall (r, s) \in E$, there is not more that one pair (r, s) that has the same point s. This restriction has an intuitive idea and this is, one point of the first curve can be matched with n points of the second one but not in the inverse sense. We have to say that the number of points of each Star Field can be different and that is because we can match polygons with different grade of evolution.

7.1 Matching raph

The main idea behind the construction of the matching graph consist in building a connected weighted graph so that an algorithm to find the minimal spanning tree is applied. The minimum spanning tree is a subset of edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. This way, the lower value of total weight the more similar are the shapes involved. But, in order to get the desired result the matching graph must be constructed in a very particular way. This method of construction is shown in the Matching graph construction algorithm .

Matching graph construction

input: two set of points SF_1 and SF_2 that define the two Star Field representations, an increment Δ and a distance d
output: a connected weighted graph

1. rotate in the x direction SF_1 and SF_2 so that, the most import star of each SF coincides in the center of the window
2. for each point sf_1pn from the SF_1 do
3. look for those points that belong to SF_2, that stay at most a distance d in all directions from sf_1pn and that have not been connected previously
4. connect sf_1pn with each point found in previous step and assign a weigh equal to the euclidian distance of the two vertices to each edge
5. if there wasn't any connection, increase d in a value Δ and go to step 3
6. Select one point of SF_1 and connect the rest of the points from SF_1 with it; finally assign each edge generated in this step a weigh equal to zero

Given two identical shapes with the same number of steps, the total weight of the spanning tree is equal to zero. This is, because each star is connected with the corresponding one and since they have the same value of $x - coordinate$ and $y - coordinate$ the euclidian distance is equal to zero. Additionally, we have mentioned that all the stars from the first shape are connected with a weight equal to zero. As a result, the values of the path through the spanning tree is zero, that means that they are identical. The algorithm for finding the minimum spanning tree most of time is called Prim's algorithm.

7.2 Similarity easure

Finally, we can define how to calculate the similarity among shapes. The most important part of this calculation is the value of the cumulative weight of the edges that make up the spanning tree. However, the similarity value is also affected by a penalty quantity, this is because some stars have not been connected with the corresponding ones.

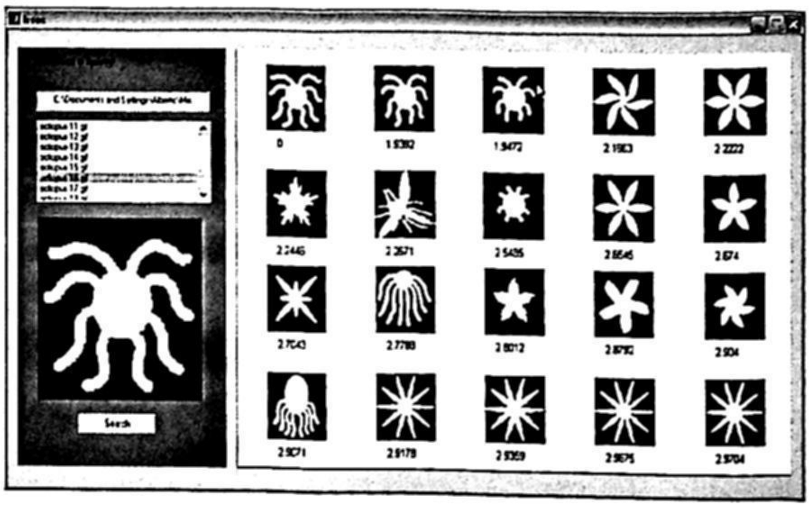

Fig. 7. IRONS' GUI, IRONS was developed using Matlab

8 Results

In the majority of the experiments of this paper we used the database CE-Shape-1 [21]. The reason why we selected this image database is because this set of images has been used for testing similar works, this allows us to have a reference framework to compare with. The Core Experiment CE-Shape-1 for shape descriptors performed for the MPEG-7 standard consists of 1400 images divided into 70 classes with 20 images each. A single image is a simple pre-segmented shape defined by their outer closed contour. Since the $2D$ objects are projections of $3D$ objects their silhouettes may change due to:

– change of a view point with respect to objects
– non-rigid object motion (e.g.people walking or fish swimming)

Table 1 describes shortly a set of shape descriptors which were tested in Core Experiment CE-Shape-1 and these works are the ones we compare with our proposed method.

First experiment consist in verifying how robust is our method with respect to scaling and rotation changes. We done this experiment in the way is described in part A of MPEG-7 standard experiments. Results are shown in table 2. Our method is labeled as **G**.

We can say that our method is robust to changes in scaling and rotation as we have already demonstrated comparing our method with those of the MPEG-7 core experiment. We cannot forget that the 91.40% was obtained in a very strict experiment and this value is not far from those reported by the the MPEG-7 core experiment and in some cases even better.

Table 1. Shape descriptors which were tested in the Core Experiment CE-Shape-1

Descriptor	Type of descriptor	Presented by	Technology
A	contour based descriptor	Mitsubishi Electric ITE-VIL	based on the curvature scale-space [19], [20]
B	contour based descriptor	Henry Hertz Institute in Berlin	based on wavelet representation of object contours [21]
C	contour based descriptor	Longin Jan Latecki and Rolf Lakämper in cooperation with Siemens Munich	best possible correspondence of visual parts [22], [23]
D	image based descriptor	Hanyang University	based on Zernike moments [24]
E	image based descriptor	Hyundai Electronics Industries	based on multilayer eigenvectors [18]
F	skeleton based descriptor	Mitsubishi Electronic and Princeton University	tree-matching algorithm [25], [26], [27]
G	**contour based descriptor**	**The authors of this paper**	**Star fields**

8.1 Similarity Based Retrieval

The retrieval rate consists in computing the number of correct matches in the top 40 retrieved images using a single image from any class as image query. There are some images in a single class that are semantically related but numerically unrelated, this is why, it is not possible to have a 100% retrieval rate. The retrieval rate of the descriptor is near to 72% table 3 shows the precise figures.

Additional experiments using our proposed technique to retrieve images as well as the IRONS system are described in detail in [4].

9 Conclusions

We proposed a complete new strategy for computing a similarity among shapes. This new technique was called Star Field (SF). Star Field inherits from $2STF$ invariant characteristics. Additionally, Star Field allows us to work with less simplified digital polygons; since, it permits to define a similarity measure based on the calculation of a minimum spanning tree from a connected weighted graph. Among the outstanding points of our set methods we can mention: ease of use and implement, it uses visual parts as a parameter of similarity like humans do, it has a good performance as we demonstrated in this paper.

The proposed technique, which is made up of a set of new methods, was implemented in a test-bed CBIR system that we called **IRONS**. IRONS stands for "**I**mage **R**etrieval based **ON** **S**hape". IRONS was developed using the Matlab

Table 2. Robustness to scaling and rotation, our results are labeled as **G**

Shape descriptor	Invariant to scaling	Invariant to rotation	Robustness to scaling and rotation
A	89.76	99.37	94.56
B	88.04	97.46	92.75
C	88.65	100.00	94.32
D	92.54	99.60	96.07
E	92.42	100.00	96.21
F	no results	no results	85
G	**91.78**	**93.05**	**91.40**

Table 3. Part B results of the Core Experiment CE-Shape-1, our proposed method is labeled as G

Shape descriptor	Similarity-based retrieval Percentage of correct matches
A	77.44
B	67.76
C	76.45
D	70.22
E	70.33
F	60
G	**71.82**

language. IRONS is just a prototype for testing our proposal image retrieval technique, and it does not pretend to be a full operational system.

To conclude, we proposed an high effective, ease to implement and robust image retrieval technique which uses the shapes of the objects as a main descriptor. Our approach is comparable in results with those systems which compute the best correspondence among shapes. However, our approach does not attend to find the best correspondence but it finds a very good approximation.

References

1. H. Blum. Biological shape and visual science. *Journal of Theor. Biol.*, 38:205–287, 1973.
2. V. C. C. and C. M. D. A review of content-based image retrieval systems. JISC Technology Applications Programme, 2000.
3. N. U. I. CC Research Laboratories. The amore, advance multimedia oriented retrieval engine. http://www.ccrl.com/amore/, 2002.

4. J. A. Chávez-Aragón. *Star Field Approach for Shape-Based Image Retrieval: Development, Analysis and Applications.* Ph.d. Thesis, Universidad de las Américas-Puebla, Mexico, 2007.
5. G. Chuang and C.-C. Kuo. Wavelet descriptor of planar curves: Theory and applications. *IEEE Trans. on Image Processing*, 5:56–70, 1996.
6. E. corp. Excalibur visual retrievalware software deveopers kit. 2004.
7. M. S. L. (Ed.). *Principles of Visual Information Retrieval.* Springer-Verlag London, 2001.
8. S. A. F. Mokhtarian and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. *A. W. M. Smeulders and R. Jain (ed), Image Database and Multimedia Search, World Scientific Publising, Singapore*, pages 51–58, 1997.
9. B. I. Recognition-by-components: a theory of human image undestanding. *Psychological Review*, 94(2):115–147, 1987.
10. V. Inc. Vir image & image read/write toolkit. http://www.virage.com/, 2004.
11. S. J. and C. S. Visual seek: a fully automated content-based image query system. *proceedings of ACM Multimedia 96*, 1:87–98, 1996.
12. H. K. and K. T. Query by visual example - content-based image retrieval. *EDBT'92 Third International Conference on Extending Database Technology*, 1:56–71, 1992.
13. S. K., S. A., and M. S. Black box approach to image feature manipulation used by visual information retrieval engines. University of Georgia, Department of Computer Science, technical report, 1997.
14. S. J. D. K. Siddiqi, A. Shokoufandeh and S. W. Zucker. Shock graphs and shape matching. *Int. J. of Computer Vision*, 2000.
15. A. Khotanzan and Y. H. Hong. Invariant image recognition by zernike moments. *IEEE Trans. PAMI*, 12:489–497, 1990.
16. M. M. Lab. Photobook. http://www-white.media.mit.edu/vismod/, 2003.
17. L. J. Latecki and R. Lakämper. Contour-based shape similarity. *Proc. of Int. Conf. on Visual Information Systems*, LNCS 1614:617–624, 1999.
18. L. J. Latecki and R. Lakämper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2000.
19. Z. Lei, R. Aibing, and Z. Aidong. Advanced feature extraction for keyblock-based image retrieval. *Proceedings of the 2000 ACM workshops on Multimedia*, 1, 2000.
20. I.-J. Lin and S. Y. Kung. Coding and comparison of dags as a novel neural structure with application to on-line hadwritten recognition. *IEEE Trans. Signal Processing*, 1996.
21. R. L. Longin J. Latecki and Eckhardt. Shape descriptors for non-ridig shape with a single closed contour. *CVPR 2000*, 2000.
22. F. M. and S. H. Query by image and video content: The qbic system. *IEEE computer*, 28(9), 1995.
23. F. Mokhtarian. Shape queries using image databases. http://www.ee.surrey.ac.uk/Research/ VSSP/imagedb/demo.html, 2002.
24. F. Mokhtarian and A. K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Trans. PAMI*, 14:789–805, 1992.
25. G. Th. and S. A. W. M. The pictoseek www image search system. *IEEE ICMCS*, 1, 1999.
26. M. M. Westphal. Imatch, mwlabs. http://www.mwlabs.de/, 2004.
27. C. Y. and K. S. Y. A hierarchical algorithm for image retrieval by sketch. *First IEEE Workshop on Multimedia Signal Processing*, 1:564–569, 1997.