2005 International Symposium on Nonlinear
Theory and its Applications (NOLTA2005)
Bruges, Belgium, October 18-21, 2005

A scale-space Reeb-graph of topological invariants of images and its applications to copyright protection

Jinhui Chao† and Shintaro Suzuki‡

†Dept. of Information and Systems Eng. Chuo University
‡Dept. of Electrical, Electronics and Communication Eng., Chuo University
1-13-27 Kasuga, Bunkyo-ku, Tokyo, 112-8551 Japan, Tokyo, 112-8551 Japan
Email: jchao@ise.chuo-u.ac.jp, shintaro@chao.ise.chuo-u.ac.jp

Abstract—In this paper, we show a Reeb-graph of topological invariants of images in a scale-space. Different from well-known scale-space trees of salient or critical points based on catastrophe or singularity theory, used in image retrieving/ processing and pattern recognition, we use topologically stable blobs or primary sketchs with nonzero lifetimes in scale and nonzero areas at each scale. The continuum of such blobs as a 3D manifold is featured by tree of topological invariants called Reeb graph. We show that this Reeb-graph representation is more robust against deformation attacks and perturbation such as numerical errors than traditional scale-space trees. A fast matching algorithm for the graph is also presented. This scale-space Reeb-graph can be applied to passive watermark, copyright tracing/monitoring and protection.

1. Introduction

Copyright protection of digital contents over internet is an important yet difficult task. To currently used watermarking, it seems that various deformation attacks are inevitable and hard to defense. These attacks include geometric such as Euclidean and Affine transforms or other nonlinear deformations. Other image processing such as compression, filtering, gray scaling or histogram transforms or other operations could also significantly reduce detectability of watermarks. On the other hand, higher detectability by stronger embedding always means quality degradation of contents.

It is for these reasons that recently, "passive watermarking" techniques under names of "image hashing", "soft authentation" etc have been observed[12],[13]. These methods, instead of embedding foreign information into contents, try to identify illegal copies of the original contents using their intrinsic features. Advantages of this method include firstly no quality deterioration of the contents, and secondly the intrinsic features of contents will not by easily erased as the presently used watermarks.

Until now most active or passive watermark methods have been reported based on local features which are defined by e.g. wavelet transforms of GOP, or meshes. However, it is more desirable to use global features in order to resist deformation attacks containing a shift farther than the size of the image blocks or GOPs, since the shift distance of an attack is usually hard to predict.

The scale space is well known as a powerful tool in image analysis and pattern recognition. Especially, it is used to reveal the "deep structure" of an image, by considering all level of scales simultaneously. It is in fact, natural to regard the 1-parameter continuum of the scale- space images as a topological object and extract its invariants. However, it seemed that among researches until now, only the linear scale space filtering method was able to extract the topology of this aggregation or stacks of the image in different levels of scale, which was a straightforward application of the singularity theory on bifurcation trajectories of the non-Morsian critical points. These properties are not general enough and basically structurally unstable.

On the other hand, a tree structure especially a scale-space tree which represents the image in a descending order of resolution or scale can be convenient in searching, e.g. in a large database. The scale-space trees are often built by nonlinear filters such as sieve methods[6][7] or critical points filter [8] [9]. As shown later, these scale-space trees unfortunately may not suitable to copyright tracing/monitoring. Another problem is that most of them use salient points as features, which is also topologically instable. In fact, many researches based their methods on edge detection, segmentation or oject recognition. Such discontinous and not well-defined operations are obviously not desirable in copyright tracing.

In this paper, we will address ourselves to a new scale-space tree called scale-space Reeb-graph. We will first review existing approaches on scale-space trees, then discuss design strategy of a good scale-space tree fro copyright tracing. The scale-space Reeb-graph is defined by specifying scale filters, blobs and tree structure. We show that this new scale-space tree are robust against deformation and noise. Finally an algorithm for fast matching of the scale-space Reeb-graph is presented. Simulation is also provided to examine the performance of the proposed method under deformation attacks.

2. Existing approaches of scale-space or multiresolution trees

Scale-space trees or multiresolution trees can be divided into the following categories.

(1) Linear (e.g. Gaussian) filtering tree

Consisted basically of trajectories of critical points of an image or certain feature of the image when the scale increased. It seemed this is the only example of "scale provides topology", which reflects merely a limited property of the scale space. Besides, the properties of non-Morsian critical points are structually unstable or topologically unstable, which means they are not robust again small pertubations.

Besides, for 1D signals, even it is proved for topological events in a discrete scale determine topology in the continuous scale at least in principle, but robusty is unknown. For 2 or higher dimensions, both robusty and discretization problems are not yet solved[5].

(2) Nonlinear filtering

Since linear scale-space filtering could blur edges or salient points in the original image, nonlinear filters are used to build a multiresolution or scale-space tree. (i) Sieve method [6][7] uses morphological filters such that it can preserve edges and could be robust against impulse noises. (ii) Critical point filters[8][9] preserve both positions and intensities of critical points in the original image. They are also based on morphological operations.

These nonlinear scale-space trees could be noise sensitive due to discontinous operations of the morphology filters. Since such a discontinous operation does not preserve topology, one can not guarantee uniqueness of the tree for an image under perturbation or attacks.

In summary, scale-space trees or multiresolution trees are mostly based on features of salient points , which are not robust under pertubation and deformation attack.

Besides, in traditional image processing, pattern recognition and image retrieving, performance under noisy environment is always a major concern. Especially, robusty against the impluse noise possesses a larger priority than smooth e.g. Gaussian noise. Target images are usually assumed to be exactly the same as the query image, and there is no assumption of any kind of deformations.

In copyright tracing or monitoring, noise is no longer a major issue but deformation is of the most important. In fact, one's main concern is about the misusage of contents in the same quality but some different forms.

These differences make the strategy for copyright tracing quite different from those in image retrieving.

3. A scale-space Reeb-graph

A desirable scale space tree for copyright tracing or monitoring should have the following features:

1. Robusty against deformation, e.g., the tree should contains no geometric information such as positions or orientations but topological invariants only.

2. Robusty against noise/error in implementation and variety of extraction methods.

3. Multilevel representation of the details at different resolutions. Minor deformation or noise/pertubation only affects the matching at low resolution levels.

4. Effiency or can be extracted easily and quickly.

Bellow, we show our strategy to build a scale-space Reeb-graph.

1. Scaling filters

It is clear that a continous operator either a linear or nonlinear filter is desirable in order to preserve topological invariants. In this paper, we use the simplest Gaussian scale linear filtering. Another reason to use Gaussian filtering is that it could be implemented by several fast algorithms [14],[15]. This could shorten the time for online extraction of the tree or for application in a large database.

2. Blobs

The blob or primary sketch in the scale space is also an important issue. Instead of structually unstable salient points, we use topologically robust features as blobs which keep appear in a scale-interval of a certain length and also has enough area at each scale-level. i.e. This means these features possess large enough 3D volume in scale-space.

In this paper, we use the blobs as the possitive areas of Gaussian curvatures of tristimulus.

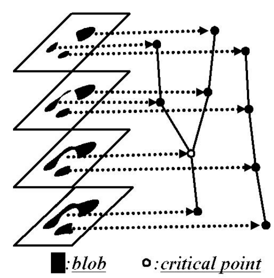

Figure 1: Blobs with nonzero areas

3. Reeb graph

The stacks of all blobs at every scale-level or the 3D continuum in the scale space we have obtained above can be regarded as a topological manifold. One way to describe such a manifold is Morse theory and can be efficiently represented by Reeb graph. [11].A Reeb graph is derived by contracting all connected components of level sets of a Morse function on a smooth manifold into a point. On these graphes, each vertex corresponds to a critical point and information of its index is also labeled on the vertex(fig.2).

We define our scale-space tree as a scale-space Reeb-graph.

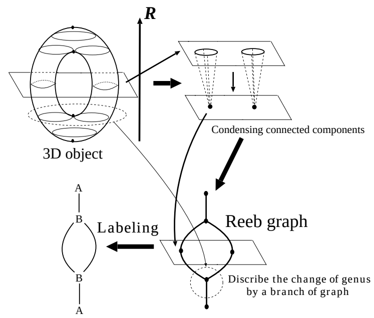

Figure 2: Reeb Graph

3.1. Construction of scale-space Reeb-graph

In fact, we will never extract the nodes in the original Reeb-graph. Since we wish to avoid topologoical instability in this extraction, only edges will be detected and recorded. According to Morse theory, this information alone will still be enough to determine the topology of the manifold. Using blobs chosen before, we can keep disgarding topologically instable features which could be affected by deformation, noise and other pertubation.

We actually also use a weighted tree, i.e. each edge is labeled with feature information such as the areas of blobs and moments of the areas. This information will be used in fast matching of these graphs.

4. Matching Algorithm

4.1. Fast matching of nodes

In order to avoid the exhaustive matching of all combinations of nodes at the same depth of the tree, the nodes are firstly matched using the informaiton of blobs corresponding to the nodes such as areas and means m, quadratic moments ($\Psi_1, \Psi_2, \cdots, \Psi_7$) etc.

$$
\begin{aligned}
\Psi_1 &= \eta_{20} + \eta_{02} \\
\Psi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{22}^2 \\
&\vdots \\
\Psi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta 12)\{(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\} \\
&\quad +(3\eta_{12} - \eta_{30})(\eta_{21} + \eta 03)\{(3\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\} \\
\text{where} \quad \eta_{pq} &= \frac{\mu_{pq}}{\mu_{00}^r} \\
\mu_{pq} &= \sum_{x \in b_i} \sum_{y \in b_i} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \\
r &= (p + q + 2)/2.
\end{aligned}
$$

Representing a node as a point in the feature space with coordinates ($m, \Psi_1, \Psi_2, \cdots, \Psi_7$) (fig.3). The two nodes r_i, r'_j matches if they are the nearest pair.

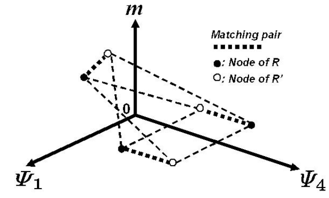

Figure 3: Matching of nodes

Remark: In order to find an embedded image, we record also matching between any subsets of the whole nodes.

4.2. Matching of whole Reeb graph

We define a similarity $\sigma(R, R')$ between two Reeb-graphes R, R' using only local topology around nodes. It is described by the subgraph of its neighborhood using only the number of edges. This is a variation of the method in [10].

$$
\begin{aligned}
\sigma(R, R') &= S(R, R')/S(R, R). \\
S(R, R') &= \sum_{r \in R, r' \in R'} s(r, r') - \frac{1}{2}\left(\sum_{\bar{r} \in R} d_{w_{\bar{r}}} + \sum_{\bar{r}' \in R'} d_{w_{\bar{r}'}}\right).
\end{aligned}
$$

The similarity $s(r, r')$ between nodes r, r' is defined as

$$
\begin{aligned}
s(r, r') &= 2^{-d_{w_r}} \left\{ w \cdot \left(1 - \frac{\|L_r - L_{r'}\|}{max(L_r, L_{r'})}\right) \right. \\
&\quad \left. +(1 - w) \cdot \left(1 - \frac{\|U_r - U_{r'}\|}{max(U_r, U_{r'})}\right) \right\}.
\end{aligned}
$$

Here $L_r, L_{r'}$ are the number of proceding nodes of r, r'. $U_r, U_{r'}$ the numbers of subnodes r, r'. w is the weight of $U_r, U_{r'}$ and $L_r, L_{r'}$. When $U_r = U_{r'} = 0, U_r = U_{r'} = 0$, one takes $max(U_r, U_{r'}) = 1, max(L_r, L_{r'}) = 1$. $\bar{r}, \bar{r'}$ are the number of unmatching nodes, d_{w_r} the depth from the node r upto the root node.

5. Simulations

The images of "Girl,Aerial,Baboon,F16,House,Lena,Milk drop,Parrot,Pepper" are used in simulation. The distrubution surfaces K_R, K_B, K_G of Gaussian curvature for the tri-stimulus R, G, B are computed. The scale-space representations or surfaces $\{L_{K_R}(\boldsymbol{x}, t)\}$, $\{L_{K_G}(\boldsymbol{x}, t)\}$, $\{L_{K_B}(\boldsymbol{x}, t)\}$ are formed by Gaussian filtering of 3×3 neighborhood. The blobs are chosen as either the negative part of the positive part of the Gaussian curnatures, then transformed to binary images. In Fig.4 only the Reeb graph of $\{L_{K_R}(\boldsymbol{x}, t)\}$ of the Girl is shown. The propose method is then applied to copyright tracing under various attacks of StirMarks4.0 . The matching results of similarity between the original and deformed "Girl" are shown in Table 1. Matching between deformed "Girl" and other images is shown in Table 2. These results shown a discrimination performance good enough for copyright tracing or monitoring.

Attack	σ	Attack	σ
JPEG:100	0.9434	JPG:90	0.9401
JPEG:80	0.9248	JPEG:70	0.9449
JPEG:60	0.9330	JPEG:50	0.9221
JPEG:40	0.9231	JPEG:35	0.9242
JPEG:30	0.9186	JPEG:25	0.9219
JPEG:20	0.9167	JPEG:15	0.9110
AFFINE:1	0.9234	AFFINE:2	0.8511
AFFINE:3	0.9470	AFFINE:4	0.8827
AFFINE:5	0.9110	AFFINE:6	0.9026
AFFINE:7	0.9324	AFFINE:8	0.9233
ROTATION:0.25	0.9521	ROTATION:0.50	0.9155
ROTATION:0.75	0.9030	ROTATION:0.90	0.8641
ROTATION:1.00	0.8724	ROTATION:5.00	0.8405
ROTATION:10.0	0.8420	ROTATION:15.0	0.8147
ROTATION:30.0	0.8024	ROTATION:45.0	0.7991
ROTATION:90.0	0.9824	RESCALE:0.50	0.8702
RESCALE:0.75	0.8901	RESCALE:0.90	0.9591
RESCALE:1.10	0.9348	RESCALE:1.50	0.9095
RESCALE:2.00	0.9016	PSNR:10	0.9849
PSNR:20	0.9837	PSNR:30	0.9753
PSNR:40	0.9803	PSNR:50	0.9618
PSNR:60	0.9728	PSNR:70	0.9801
PSNR:80	0.9845	PSNR:90	0.9797
PSNR:100	0.9814	-	-

Table 1: Similarity:"Girl" - attacked ones

Image	σ	Image	σ
Aerial	0.4036	Baboon	0.5721
F16	0.3757	House	0.4905
Milk drop	0.4210	Lena	0.7134
Parrot	0.4454	Pepper	0.5185

Table 2: Similarity:"Girl" - other images

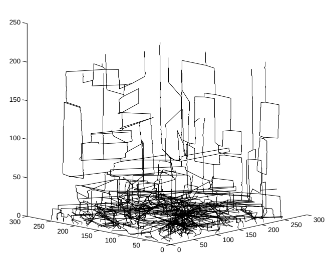

Figure 4: Reeb Graph of curvature of Red

References

[1] M. Suzuki, J. Kim, J. Chao Proceedings of 2002 Symposium on Cryptography and Information Security, SCIS2002. pp.903-907.Jan.2002

[2] M. Suzuki, H.Tamakawa, M. Makino, J. Chao Proc. Nolta2000, Dresden, Vol. 2, pp. 603-607, Oct. 2000

[3] J. Kim, J. Chao Proceedings of 2003 Symposium on Cryptography and Information Security,SCIS2003.pp959-964,Jan.2003

[4] T. Linderberg: Kluwer Academic Publishers.

[5] L.Florack and A.Kuijper Journal of Mathematical Imaging and Vision, 12, 65-79, (2000)

[6] J.A. Bangham, P. Chardaire, C.H. Pye, P.D. Ling IEEE Transactions on Pattern Analysis and Machien Intelligence, 18 (5): 529-539, 1996.

[7] R. Harvey, J.A. Bangham and A. Bosson In Proc. First Int. Conf. on Scale-space theory". P.341-344, Springer, 1977.

[8] Y.Shinagawa, T.Kunii IEEE Transactions on Pattern Analysis and Machine Intellegence, Vol. 20, No. 9, September, 1998.

[9] K.Habuka and Y.Shinagawa International Journal of Computer Vision, 58 (1), 19-35, 2004.

[10] M.Hilaga, Y.Shinagawa, T.Kohmura, T.L. Kunii Proc. ACM SIGGRAPH 2001.

[11] A.T.Fomenko,T.L.Kunii, Springer-Verlag,1997.

[12] C.S.Lu,C.Y.Hsu,S.W.Sun,P.C.Chang, Proceeding of ICME2004,Taiwan,2004

[13] B.B.Zhu,M.D.Swanson,A.H.Tewfik,IEEE Signal processing magazine, Vol.21,No2,pp40-49,Mar.2004

[14] L.J.van Villet,I.T.Young,Signal proc.,vol44,pp.139-151,1995.

[15] L.J.van Villet,I.T.Young,P.W.Verbeek,Proceeding ICPR1998,Australia.