VIDEO SEGMENTATION USING STABILIZED INVERSE DIFFUSION

Ankur Mani
Motorola India Electronics Ltd., Hyderabad
ankurmani@motorola.com

ABSTRACT

A video segmentation scheme using stabilized inverse diffusion equations is proposed. The scheme is based on the non-linear scale-space analysis of video lattice. Non-linear diffusion and especially Stabilized inverse diffusion is a very robust method for image segmentation. In this paper the two-dimensional diffusion process has been extended to three dimensions. The segmentation scheme is based on this extended process. Also, a progressive scale-space evolution of the image sequence is proposed that reduces the memory requirements as well as solves the region correspondence problem. Finally, segmentation results for two image sequences are shown.

I INTRODUCTION

Video segmentation and object extraction has for long been an area of research. With the advent of object-based multimedia and the development of standards such as MPEG-4[1], the necessity of segmenting the video into semantically defined objects has increased. A large number of schemes for segmentation of video have been proposed, some very complex, using multimodal features like color, texture and motion either iterative or parallel [2]. Most of these algorithms are based on segmentation of a reference frame and tracking of regions through successive frames, refining the original estimates with the motion information. The segmentation of the reference frame and the tracking of the regions have been achieved well using the non-linear scale-spaces and morphology.

Non-linear scale-spaces have been the established approach for image simplification and segmentation. Even for video segmentation and motion tracking, methods based upon morphology(mostly based on watershed transforms)[3,4] and anisotropic diffusion [5] have been used.

However, very few approaches have actually tried to make use of the scale-spaces of video lattices. Some segmentation of video lattices using 3D watershed transforms have been proposed however[6]. The advantage of using such an approach is that motion information is embedded in 3D segments and separate motion estimation and compensation strategies need not be employed. The problems with using the three-dimensional segmentation of video lattices arise due to the fast moving objects and due to large memory requirement.

In this paper, a new approach towards video segmentation and object extraction is introduced. The algorithm uses a family of first order Ordinary Differential Equations (ODE) stabilized at local extrema, Stabilized inverse diffusion equations (SIDE) [7, 8] to form scale-spaces of video lattice (image sequence put together to form a lattice with time being the third dimension).

This approach in already proven in image segmentation and has been recognized because of its spatial consistency and crisp segmentation results. In this paper the two-dimensional image segmentation approach has been extended to three-dimensional segmentation of video lattices. An overlapping clustering scheme for clustering frames to form a video lattice and a scene change detection scheme has also been suggested that reduces the number of frames needed to be processed at a time (thus reducing memory requirements) and at the same time maintains the correspondence between the regions in a scene.

The organisation of the rest of the paper is as such. In section 2 the stabilized inverse diffusion process and the dynamics of the three-dimensional inverse diffusion process has been discussed. In section 3, the algorithm for the video segmentation and mask generation has been introduced. In section 4 the results are discussed. Section 5 talks about current limitations of the algorithm and finally conclusions and future work are discussed in section 6.
II STABILIZED INVERSE DIFFUSION EQUATION

Stabilized inverse diffusion equations (“SIDE”) introduced by Pollak et. al. [7, 8] may be considered as a limiting case of a semi-discretized Perona-Malik equation [9, 10]. They are a semi-discrete version of PDE (continuous in time and discrete in space) that result in a system of ordinary differential equations (ODEs) with a discontinuous driving force [7, 8]. A robust and efficient implementation of an image segmentation scheme based on SIDEs using region adjacency graph is discussed in [11]. Here we extend the two dimensional SIDEs to three dimensions for the scale-space analysis of the video lattice.

Let the video lattice be represented as \( u(x,y,z) \), where \( x, y \) and \( z \) stand for the horizontal and vertical dimensions of the images and the additional time dimension of the lattice. The scale-space of the lattice is represented as \( u(x,y,z,t) \), where \( t \) stands for the scale. The force function ‘\( F \)’ controls the dynamics of the state-space of \( u \). The dynamics of a lattice point is controlled by the following equation

\[
\frac{du_{k,l,m}}{dt} = \frac{1}{m_{k,l,m}}(F(u_{k+1,l,m} - u_{k,l,m}) + F(u_{k,l+1,m} - u_{k,l,m}) + (F(u_{k,l+1,m} - u_{k,l,m}) + F(u_{k,l+1,m} - u_{k,l,m})) + (F(u_{k,l+1,m} - u_{k,l,m}) + F(u_{k,l+1,m} - u_{k,l,m}))
\]

where \( u_{k,l,m} \) denotes the gray-value of the region \((k, l, m)\) and \( m_{k,l,m} \) denotes the mass of the particle. The above dynamical system can be easily visualized with the help of a three-dimensional spring-mass model as shown in figure 1. Each node in the three-dimensional model is a region of mass \( m_{k,l,m} \), depending on the number of voxels in the region and the springs denote the force or interaction between two regions.

As the evolution progresses and the regions grow, the mass of the region (sum of the masses of the particles in the region) increases and the force on the region is equal to the sum of the forces from all the neighbors. The force function should be continuously decreasing and should have a discontinuity at \( v=0 \) [12]. The force function chosen here is

\[
F(v) = 4*\text{sign}(v)\exp(-|v|/0.25).
\]

This force function provides a sharp initial slope that prevents diffusion across the edges.

The scale-space evolution, if left unrestricted, will annihilate the whole lattice into one region. To prevent this we use the stopping criterion same as the one introduced in [11]. For that the energy function \( E_t \) is calculated as such.

\[
E_t = \sum (|\nabla u_t| + \beta(u_t - u_0))
\]

where \( u_t \) denotes the lattice at scale \( t \) and \( u_0 \) denotes the initial lattice. The summation is over the whole lattice. As can be seen, as the iterations proceed, the energy function tends to decrease because of the first term, until at some point, depending on \( \beta \) (parameter supplied to the algorithm), it suddenly starts increasing. At this point the iterations are stopped. This stopping criterion takes into consideration, both the spatial differential energy (first term) as well as the differential scale energy (second term).

III THE ALGORITHM

The algorithm is based upon the 3D non-linear diffusion process discussed above. The algorithm is summarized as shown in figure 2. The sub blocks are described further.
Preprocessing
Since the segmentation is based upon the discontinu-
ities in the gray-level or illumination, the preprocessing
step enhances the contrast of the image by histogram
equalization. Also, the camera motion can be compen-
sated as in [13].

Clustering of a group of frames
The image sequence is grouped into clusters of ‘n’
frames (this is a parameter supplied to the algorithm) to
form a lattice. In order to maintain the correspondence
between regions the clusters are made overlapping, i.e.-
the last n/2 frames of one cluster are the first n/2 frames
of the next cluster. This helps in using the segmentation
in previous cluster as a seed that propagates the segmen-
tation in the current cluster. This also helps in solving
the correspondence between regions in successive clusters.
However, when a change of scene is detected, the cluster
involving a scene change is broken into two clusters at
the change point. The detection of scene change is dis-
cussed later.

SIDE evolution and evolution stopping
The scale-space evolution and segmentation of the
lattice into regions is done using the stabilized inverse
diffusion process discussed in the previous section. This
makes three-dimensional structures for the different
regions in the sequence.

Merging of small regions
The small regions, less that ‘N’ number of pixels
(this threshold is given as a parameter to the algorithm) in
the final lattice are removed by merging them with the
most appropriate neighbors. The most appropriate neigh-
bor is the one for which the force between the current
region and the neighbor weighted by the mass of the
neighbor is maximum. This is an approximation for the
SIDE evolution, if the evolution was left to grow uninter-
rupted.

Scene change detection
A scene change is detected if in the lattice there is a
frame in which there is large discontinuity of regions, i.e.- if the sum of the number of regions disappearing and
the new regions appearing is more than a certain thresh-
old $R_{min}$ (this is a parameter supplied to the algorithm). If
no scene change is detected, the last n/2 frames of the
evolved lattice ($u(x,y,z; t_{final})$; $z \in (n/2, n-1)$) are clustered
with the next n/2 frames of the sequence and the SIDE
evolution is restarted for the new cluster. Thus the older
evolution serves as a seed in the new evolution. If a scene
change is detected the change point is noted and the steps
described further are performed.

Generation of motion parameters
Once a scene change is detected, all the evolved frames
since the last scene change are grouped together as a
macro lattice. This macro lattice has a set of parallelepi-
ped regions with the same label. For each region a set of
motion functions are identified. Currently, only the devia-
tion of the center of mass of the cross sections of regions
in each frame is considered that gives a set of vectors
describing the horizontal (dx(z)) and vertical displace-
ment (dy(z)) along the sequence. More sophisticated
motion functions based upon the motion parameters such
as in Cost 211 framework [14] can also be developed.

Object labeling and mask generation
Once the regions have been segmented, the neigh-
bor regions with similar motion are merged. The simi-
arity between the motions of two regions is estimated by
the mean–square motion deviation between the regions,

$$
\sum_{z=0}^{T} \frac{T}{2} \left( (dx_{r1}(z) - dx_{r2}(z))^2 + (dy_{r1}(z) - dy_{r2}(z))^2 \right)
$$
where \( T \) is the number of images in the scene (number of images in the macro lattice). The summation is over time. The couple or regions for which the mean-square motion deviation is less than a certain threshold \( M \) (parameter supplied to the algorithm) are merged and the motion of the combined region readjusted. The regions without any motion can be merged with the background. Once the regions are merged into objects, the 3-D object masks are extracted. After this, the whole cycle is repeated for the next scene.

**IV RESULTS**

The segmentation results for the tennis sequence and an artificial image sequence are shown in figure 3 and 4 respectively. In figure 4, the mask of the object composed of a rectangle and a circle, moving along the principal diagonal is shown. The scenes have enough noisy regions but the diffusion based segmentation process extracts the regions suitably. The segmentation is very crisp and well localized. The algorithm was implemented on a PentiumIII 450 MHz machine where it takes roughly 30 seconds for a sequence with 2 million lattice points.

**V LIMITATIONS**

The scheme and the system is still developing and a lot of additions have to be done to make it a practical system for segmenting real image sequences. Currently, only rigid objects are considered for segmentation. Also, the center of mass of the background develops some motion error due to the moving objects in the scene. These issues will be solved by using sophisticated motion models. Also, the algorithm currently takes gray-level as the only feature for the diffusion process. Currently, there is no method to detect the occlusions and the emerging new regions in the scene.

**VI CONCLUSION**

The novel scheme for spatio-temporal segmentation of video has been introduced in the paper. The algorithm works well for video with moderate motion. The future work is aimed at making the system more robust and also enhance the system to tackle with the occlusions and emerging new regions. An adaption in the algorithm to take the advantages of multiple features such as luminance and chrominance for the segmentation of vector-valued images is being developed based on [15]. The system is efficient enough due to progressive clustering. Future work also aims at further improving the performance of the algorithm and contour and texture coding of the objects.

**REFERENCES**


