DWT BASED HIERARCHICAL VIDEO SEGMENTATION

Sunder Venkateswaran and U. B. Desai

Department of Electrical Engineering
Indian Institute of Technology, Bombay
Powai, Mumbai-400076, India
Email: ubdesai@ee.iitb.ac.in

ABSTRACT

To provide multimedia applications with new functionalities, such as content-based interactivity and scalability, the new video coding standard, MPEG-4 relies on content-based representation. This means a prior decomposition of a video sequence into semantically meaningful, physical objects. We develop a fast yet robust color segmentation algorithm based on DWT and k-means clustering. After spatial segmentation, a dense optical flow field is estimated for the whole image, using which a motion parameter model is estimated for each region using a least squares technique. Adjacent regions with a coherent motion are then merged together to form moving objects. This leads to segmented video.

1. INTRODUCTION

Content-based video coding methods describe the video in terms of objects which are homogeneous in terms of color, intensity, motion etc. The new MPEG-4 video coding standard [1] is based on such an object-based representation of video. The first step in content-based video coding is the video object segmentation process which decomposes the scene into semantically meaningful objects, also called video object planes (VOPs).

The first image of the sequence is segmented into regions that are homogeneous in terms of color, intensity or texture. This is the spatial segmentation step. To extract objects, a motion model is estimated for each of these spatial segments. Since a moving object may consist of adjoining regions which have similar motion, adjacent segments with a coherent motion are merged to form moving objects.

A number of techniques and algorithms for video segmentation have been proposed. In [2], consecutive frames in a video sequence are spatio-temporally segmented, independently of each other. The approach presented in [3], [4] deals with the sequence as a 3D (2D plus time) signal and, therefore, performs a 3D segmentation. In [5], motion projection is used for temporal tracking. Motion projection may result in uncovered and overlapping regions. These regions are considered as new regions.

Color has been used as the main feature in image segmentation and content-based video query. The approach by Zhong and Chang [6] separates images into homogeneous regions using color segmentation, after which they are tracked in time.

In this paper, we present a fast and robust color segmentation method based on the discrete wavelet transform (DWT) and the k-means clustering algorithm, which is first applied in the spatial domain. With the usual K-means clustering method, it is difficult to determine the initial number of classes. Hence we use a modified version of the same, where a simple validity measure based on the intra-cluster and inter-cluster distance measures allows the number of clusters to be determined automatically. After dividing the image into homogeneous spatial regions, a dense optical flow field is estimated for the whole image. A six parameter motion model is estimated for each region from the optical flow field using the least squares method. After this adjacent regions with a coherent motion are merged together to form moving objects.

2. SPATIAL SEGMENTATION

2.1. The Color Clustering Algorithm

Clustering is one of the simplest and widely used methods in the segmentation of gray-level and color images. The k-means algorithm requires the number of clusters to be known beforehand. We have used the method proposed by Ray and Turi [8], which overcomes the limitation of having to indicate the number of clusters by incorporating a validity measure based on the intra-cluster and inter-cluster distance measures.

The usual K-means method aims to minimize the sum of squared distances between all points and the cluster center. This procedure consists of the following steps.

1. Choose K initial cluster centers, \( c_1(1), c_2(1), \ldots, c_k(1) \).
2. At the k-th iterative step, distribute the samples \( \{x\} \) among the K clusters using the relation,

\[
x \in C_j(k) \text{ if } \|x - c_j(k)\| < \|x - c_i(k)\|
\]
for all \( i = 1, 2, \ldots, K \) where \( C_j(k) \) denotes the set of samples whose cluster center is \( c_j(k) \).

3. Compute the new cluster centers \( c_j(k+1) \),
\( j = 1, 2, \ldots, K \) such that the sum of the squared distances from all points in \( C_j(k) \) to the new cluster center is minimized. The measure which minimizes this is simply the sample mean of \( C_j(k) \). Therefore, the new cluster center is given by
\[
c_j(k+1) = \frac{1}{N_j} \sum_{x \in C_j(k)} x, \quad j = 1, 2, \ldots, K
\]
where \( N_j \) is the number of samples in \( C_j(k) \).

4. if \( c_j(k+1) \approx c_j(k) \) for \( j = 1, 2, \ldots, K \) (within a threshold), then the algorithm has converged and the procedure is terminated;
otherwise go to step 2.

We briefly describe the validity measure proposed in [8] to estimate the number of clusters. Since the K-means method aims to minimize the sum of squared distances from all points to their cluster centers, this should result in compact clusters. Therefore the distances of the points from their cluster center can be used to determine whether the clusters are compact. For this purpose, the intra-cluster distance measure is used, namely,
\[
intra = \frac{1}{N} \sum_{i=1}^{K} \sum_{x \in C_i} ||x - c_i||^2
\]
where \( N \) is the number of pixels in the image, \( K \) is the number of clusters, and \( c_i \) is the cluster center of \( C_i \). Obviously this measure has to be minimized. The inter-cluster distance or the distance between clusters, which we want to be as big as possible, can be calculated as follows,
\[
inter = \min(||c_i - c_j||^2), \quad i = 1, 2, \ldots, K-1 \quad j = i+1, \ldots, K
\]
In order to use both these measures to determine if we have a good clustering, a validity measure is defined as the ratio of the intra and inter cluster distances.
\[
validity = \frac{\text{intra}}{\text{inter}}
\]

Thus, the clustering which gives a minimum value for the validity measure will tell us what the ideal value of \( K \) is in the k-means procedure.

2.1.1. Description of the proposed clustering method

A number of color spaces exist in which segmentation of images can be performed. Here, we have used either the RGB or the YCbCr color space.

Segmented images from 2 up to \( K_{\text{max}} \) clusters are produced, where \( K_{\text{max}} \) is an upper limit on the number of clusters, and the validity measure is calculated at each stage to determine the best clustering, and, therefore, the optimal value of \( K \).

We do this by first computing a two-level wavelet decomposition of the image [7]. We have used the daub-4 wavelet for this purpose. Once this is done, we first form one cluster containing all the pixels in the LL image at the lowest resolution (i.e. at level 2). Then an iterative process begins where, unless the number of clusters is equal to \( K_{\text{max}} \), the cluster having maximum variance is split into two. Once the cluster is split, the k-means procedure is used to obtain the clustering for this new number of clusters. Once all the clusters have been formed, the validity measure can be calculated for each of them to determine the optimal value of \( K \).

Since the K-means algorithm aims at minimizing the average intra-cluster distance, it is most likely that the cluster having maximum variance will be separated by the K-means procedure when the number of clusters is increased. Therefore, when the number of clusters has to be increased, the cluster having maximum variance is split, so that the K-means procedure is given good starting cluster centers. In the case of natural images, there is a tendency for the minimum of the validity measure to occur for a small number of clusters (in the range of 2, 3 to 4). This is due to a large inter-cluster distance value occurring when the number of clusters is low, resulting in the validity measure being very small. In general natural color images will have greater than 2, 3 or 4 clusters. So, instead of simply selecting the clustering which leads to the minimum value of the validity measure, the first local maximum in the validity measure is found, where a local maximum is defined to occur at \( k \) if
\[
validity(k-1) < validity(k) > validity(k+1), \quad k \neq 3
\]
The minimum after this first maximum is chosen.

Once the clustering is done for the image at the lowest resolution, the centers obtained are used as initial seeds for the LL image at the next higher resolution (i.e. at level 1). Since the clustering was already done at a lower resolution, the centers obtained from the previous part will be good starting points for clustering in the current resolution. Also the number of clusters can be kept fixed as there will be no new regions in the current resolution that weren’t present at the lower resolution. Here, the usual K-means method is applied as the number of clusters is already fixed. Because of good initial guesses the algorithm converges in 2-3 iterations. In this way, information is propagated from one resolution to another, till we reach the level of the original image. The above procedure can be generalized to arbitrary number of levels. Since the modified K-means method is applied only to the image at the lowest resolution and not the original image, the overall time taken for the clustering
is considerably reduced.

3. MOTION TRACKING AND MOTION-BASED REGION MERGING

Since most semantic objects are characterized by a coherent motion pattern, which is distinct from that of the background, motion is commonly used to group regions into objects. Parametric models can be applied to describe the motion of each region by a set of parameters that is estimated by fitting a model in the least-squares sense to a motion field obtained by a non-parametric method, such as optical flow. Among several parametric models, the affine and perspective motion models are the most frequently used. Once the motion parameters of a region are obtained, the region can be tracked in subsequent frames.

3.1. Motion Estimation

We have used the approach proposed by Black and Anandan [9] for dense and robust optical flow estimation. This robust formulation, combined with a deterministic optimization scheme, allows the brightness constancy and spatial smoothness assumption violations to be detected.

3.2. Motion-based Region Merging

Adjacent regions with coherent motion should be merged together to form moving objects. The similarity between two regions in terms of motion is measured by the increment of the mean-square motion compensation error [10]. Region merging is realized in the following steps.

1. Motion parameters are estimated for every region $R_i$. For each region the sum of squared compensation error $E_i$ is calculated, which is given by:

$$E_i = \sum_j \left[ (d_x - u_j)^2 + (d_y - v_j)^2 \right]$$

Here $j$ ranges over all the pixels $(x_j, y_j)$ in the region $R_i$. $u_j$ and $v_j$ are the $x$ and $y$ components of the optical flow vector at pixel $(x_j, y_j)$. $(d_x, d_y)$ is the motion vector at pixel $(x_j, y_j)$ computed from the motion parameter model.

2. For every two adjacent regions $R_i$ and $R_j$, a set of motion parameters is estimated for the regions combined together ($R_i \cup R_j$). Let $E_{i,j}$ denote the sum of squared compensation error of the two regions when they are compensated using this set of motion parameters. The increment of mean square motion compensation error is calculated by $\Delta_{i,j} = (E_{i,j} - E_i - E_j)/(N_i + N_j)$, where $N_i$ and $N_j$ are the number of pixels in regions $R_i$ and $R_j$, respectively.

3. If the value of $\Delta_{i,j}$ is smaller than a predefined threshold, then the corresponding regions are merged.

4. All of the $E_i$, $\Delta_{i,j}$, and $N_i$ which are related to the merged other regions until every $\Delta_{i,j}$ is greater than the threshold.

3.3. Segmentation Of Subsequent Frames

Moving objects are tracked in the next frame according to their affine motion models. In the meantime, the next frame is also segmented into homogeneous color regions with the technique presented in Section 4.1. If 75% of the pixels are common between two regions of two consecutive frames, we say that these two regions are matched and treat them as corresponding regions. Therefore, tracked region boundaries can be updated by those of the corresponding regions based on spatial segmentation. For unmatched regions, change detection is used to find moving regions. This process allows detection of newly appearing objects in the scene.

4. SIMULATION RESULTS

We tested the algorithm on the "Akiyo" QCIF MPEG-4 test sequence. The daub-4 wavelet was used for wavelet decomposition. The K-means clustering was then performed in the RGB color space of the image at the lowest resolution using the Mahalanobis distance measure. The value of $K_{max}$ was set at 15. The final number of clusters for which the validity minimum was reached, was found to be 9. Then using these cluster centers as initial seeds, clustering was performed in the color space of the image at the next higher resolution. The number of clusters was kept constant.

Since the clustering was performed in the color space, the result contains regions which have the same label but are disconnected. To give each connected region a distinct label, a region separation operation is necessary. Regions which are smaller than 0.5% of the image size are merged into a neighboring region to which they are closest in the mean. The result obtained after these two operations is shown in Fig. 1.

The next step is motion-based region merging. An optical flow field is first estimated using the robust method of Black and Anandan [9]. Adjacent regions with a coherent motion are merged to form moving objects. After this, the resulting segments are semantically meaningful. The result of motion-based region merging for the first frame of the "Akiyo" sequence is shown in Fig. 2.

After the segmentation of the first image, its cluster centers are used as initial guesses for clustering the next image. Again the number of clusters is kept constant. Here the clustering is performed on the pixels of the entire image, and since the K-means procedure is given good starting cluster centers, the algorithm converges in 4 or 5 iterations. Then
the regions are merged according to motion information to form moving objects.

The segmentation of seven frames of the “Akiyo” QCIF sequence is shown in Fig. 3. Due to space limitations, we have depicted the segmentation of only three frames in Fig 3. We see that each image is segmented into a few meaningful regions like hair, facial region, shoulder and the background.

5. REFERENCES


