Video Segmentation: Propagation, Validation and Aggregation of a Preceding Graph

Siying Liu[1], Guo Dong[2]
[1,4]National University of Singapore
Electrical and Computer Engineering
4 Engineering Drive 3, Singapore 117576
[1]elels@nus.edu.sg [4]eleongsh@nus.edu.sg

Chye Hwang Yan[3], Sim Heng Ong[4]
[2,3]DSO National Laboratories
20 Science Park Drive, Singapore 118230
[2]gdong@dso.org.sg [3]ychyehwa@dso.org.sg

Abstract

In this work, video segmentation is viewed as an efficient intra-frame grouping temporally reinforced by a strong inter-frame coherence. Traditional approaches simply regard pixel motions as another prior in the MRF-MAP framework. Since pixel pre-grouping is inefficiently performed on every frame, the strong correlation between inter-frame groupings is largely underutilized. We exploit the inter-frame correlation to propagate trustworthy groupings from the previous frame. A preceding graph is constructed and labeled for the previous frame. It is temporally propagated to the current frame and validated by similarity measures. All unlabeled subgraphs are spatially aggregated for the final grouping. Experimental results show that the proposed approach is highly efficient for spatio-temporal segmentation. It makes good use of temporal correlation and produces satisfactory grouping results.

1. Introduction

Video segmentation is used in various vision applications. The exact meaning of the term video segmentation varies according to the context in which it is used. It refers to a decomposition of semantic entities in content-based video retrieval [4] and video epitomes, a segmentation of moving blocks in video coding or a spatio-temporal grouping in scene interpretation [8, 13], etc. In this paper, we address it as an efficient intra-frame segmentation reinforced by inter-frame coherence. It is a problem of pixel labeling based on temporal coherence and spatial grouping.

Given an image of N pixels, let $\mathbf{S} = \{s_1, s_1, \ldots, s_N\}$ be a set of image pixels. Define $\mathbf{X} = \{X_s | s \in \mathbf{S}\}$ as a family of random variables, and $\mathbf{L} = \{1, \ldots, l_M\}$ as a set of label states. To segment the image into l_M perceptual groups, each pixel is assigned one of the prescribed labels l_m so that $\forall s \in \mathbf{S}, X_s \in \mathbf{L}$. Using only constraints from image data, it is an ill-posed problem. With the prior distribution of image labels, Bayes' rule is a principal way that best estimates the likelihood of image labels by

$$P(\mathbf{X}|\mathbf{S}) \propto P(\mathbf{S}|\mathbf{X})P(\mathbf{X}) \quad (1)$$

Image labeling is the maximum a posteriori (MAP) estimation of $P(\mathbf{X}|\mathbf{S})$. In the MRF-MAP framework, $P(\mathbf{S})$ is modelled as a Markov Random Field (MRF), which allows the incorporation of contextual constraints based on piecewise constancy [6]. Using a log likelihood of $P(\mathbf{X}|\mathbf{S})$, MRF-MAP is equivalent to the regularization of $\mathbf{X}$ by minimizing the energy function

$$E = E_d + \lambda E_s \quad (2)$$

where E_d is the energy of image data, E_s is the smoothness energy, and λ is a weighting factor. The elegance of MRF-MAP framework simplifies the image segmentation problem as an exact minimization of the above energy equation by seeking a global solution for a non-convex energy in a high dimensional space. Unfortunately, such an approach is known to be difficult due to a large number of local minima.

1.1. Graph Cut

Image segmentation can intuitively be viewed as an optimal cut of graph $\mathbf{G} = (\mathbf{S}, \mathbf{E})$, where $\mathbf{S}$ is the set of image pixels and $\mathbf{E}$ is the set of edges connecting to neighboring pixels. A weight is associated with each edge based on some attributes of the pixels it connects. Considering the set of label states $\mathbf{L}$ as the terminals of graph $\mathbf{G} = (\mathbf{S}, \mathbf{E})$, the minimization of MRF-MAP is equivalent to finding a minimum cost of a multi-way cut for a graph, depending on some predefined label seeds in the image. With two terminals of source s and sink t, the Potts energy model of equation (2) can be exactly solved by a min-cut/max-flow of the s-t graph, i.e. searching the maximum flow from s to t in Ford-Fulkerson algorithm [1]. The NP-hard problem in the multi-way cut is approximated by the α-expansion algorithm. In spectral graph partitioning, the cost of bipartition

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

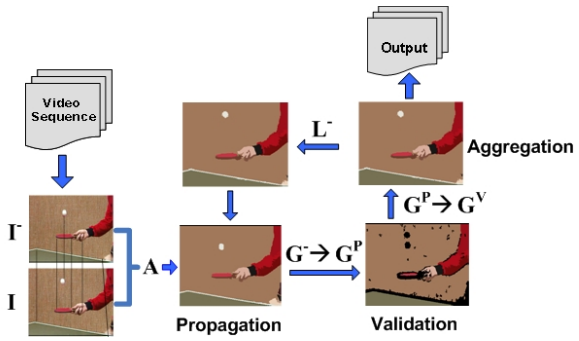


Figure 1. Spatio-temporal grouping by the propagation, validation and aggregation of a preceding graph

-ing $\mathbf{G}$ into subgraphs $\mathbf{A}$ and $\mathbf{B}$ is the sum of the weights of all edges connecting the two subgraphs, denoted as $cut(\mathbf{A}, \mathbf{B})$. The minimization of $cut(\mathbf{A}, \mathbf{B})$ is an NP-complete problem. Relaxing the membership indicator from discrete to continuous values is equivalent to solving the eigen system $\mathbf{Lx} = \lambda\mathbf{x}$ (where $\mathbf{L}$ is the Laplacian matrix of G). According to the Rayleigh quotient theorem, the minimum value of $cut(\mathbf{A}, \mathbf{B})$ is given by the second smallest eigenvalue of $\mathbf{L}$. The eigenvector λ_2 (the Fiedler vector) is the optimal solution of $cut(\mathbf{A}, \mathbf{B})$. The minimum cut criteria is prone to cutting small isolate sets. More reasonable cut criteria have been studied, including the ratio cut, normalized cut and min-max cut, etc. The min-max cut is able to perform more compact and balanced results for strongly overlapped clusters. The spectral graph cut has a high computational cost. For example, it is proportional to $O(N^{3/2})$ in the normalized cut, limiting its application on very large images. The Algebraic Multigrid (AMG) [11] is able to recursively achieve the minimization of N-cut by an adaptive graph coarsening with a computational cost of only $O(N)$.

1.2. Spatio-Temporal Grouping

Inter-frame correlation provides strong constraints for an optimal intra-frame grouping. Numerous spatio-temporal segmentation approaches have been reported in the literature [8]. Many extend the MRF-MAP framework in time and treat temporal correlation as another prior. In this case, Bayes' rule is extended to

$$P(\mathbf{X}|\mathbf{S}, \mathbf{T}, \mathbf{X}^-, \mathbf{S}^-) \propto P(\mathbf{S}|\mathbf{X}, \mathbf{T}, \mathbf{X}^-, \mathbf{S}^-)P(\mathbf{X}^-|\mathbf{X}, \mathbf{T})P(\mathbf{X}|\mathbf{T}) \quad (3)$$

where $\mathbf{X}^-$ and $\mathbf{S}^-$ denote the sets of pixel labels, and image pixels in the previous frame. $\mathbf{T}$ refers to inter-frame pixel displacements. The MRF-MAP estimation is the minimization of the energy function

$$E = E_d + \lambda E_s + \mu E_t \quad (4)$$

where λ and μ are weighting factors for smoothness and temporal coherence. This energy minimization has been suggested and solved by Iterative Conditional Modes (ICM) in [9, 13]. Under this framework, an over-segmentation has to be performed on each frame followed by enforcement of temporal coherence. Unfortunately, this approach tends to under-utilize the strong temporal correlation. Temporal correlation can be more efficiently exploited in video-based applications, *e.g.* [5]. Furthermore, MRF-MAP in (4) searches for an optimal combination of subgroups with different spatial scales. Such an approach will lead to an intractable problem of finding a model to handle variations in spatial scales. Simple pixel-based measures, such as intensity or color, are insufficient to characterize the subgroups with large scales. High-level scale measures, i.e. texture or shape, have to be incorporated since scale variations commonly occurs among the segmented subgroups [11]. Lastly, an estimation of pixel displacement has been a challenge. This is especially true for those pixels with independent motions. With an erroneous motion prior, MRF-MAP in (4) can lead to extremely sub-optimal groupings.

2. System Overview

Figure 1 shows the structure of the segmentation system. Instead of enforcing an unrectified motion prior in MRF-MAP, we propagate pixel labels from the previous frame to the current using a global motion estimation, followed by validation based on similarity measures. We preserve trustworthy propagated pixel labels, while removing erroneous labels to reduce the bias in the final grouping. All unlabeled pixels are initially grouped into subgraphs by a simple color clustering. These subgraphs are iteratively aggregated by a pairwise subgraph grouping.

This paper is organized as follows. In Section 3, the proposed approach is formulated in detail. The propagation and validation of a preceding graph is described in Section 3.1-3.3. The subgraph aggregation is specified in 3.4, followed by the analysis of computational complexity. The spatio-temporal grouping algorithm is summarized in Section 3.5. Experimental results and performance evaluations are given in Section 4 and finally, Section 5 concludes the paper.

3. Problem Formulation

A preceding graph in the previous frame $\mathbf{I}^-$ can be specified by $\mathbf{G}^- = (\mathbf{S}^-, \mathbf{E}^-, \mathbf{L}^-)$, where $\mathbf{S}^-$ is the set of all nodes (pixels), $\mathbf{E}^-$ is the set of edges connecting the nodes, and $\mathbf{L}^- = \{1, \ldots, l_M\}$ is the set of pixel labels. Temporal propagation through an affine estimation compensates global motion between two consecutive frames, thereby propagating $\mathbf{G}^-$ into $\mathbf{G}$ of the current frame $\mathbf{I}$. Let $\mathbf{G}^p = (\mathbf{S}^p, \mathbf{E}^p, \mathbf{L}^-)$ be the propagated graph from $\mathbf{G}^-$. $\mathbf{S}^p$ includes all nodes that can be projected to $\mathbf{I}$, $\mathbf{S}^p \subseteq \mathbf{S}^-$. $\mathbf{E}^p$ is the set of edges connecting the nodes in $\mathbf{S}^p$ in $\mathbf{I}$.

Considering the inter-frame pixel variations, all nodes

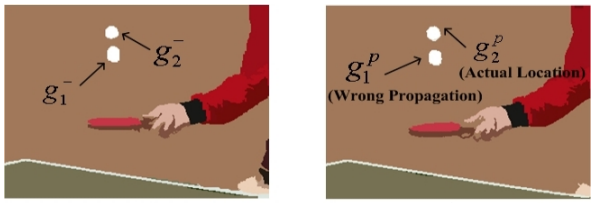


Figure 2. A pair of invalidated labels of a single region due to independent motion.

in $\mathbf{S}^p$ are validated by measuring the color similarity between $\mathbf{I}^-$ and $\mathbf{I}$. A validated graph $\mathbf{G}^v = (\mathbf{S}^v, \mathbf{E}^v, \mathbf{L}^{\mathbf{V}})$ is formed by removing the nodes with wrong labels from $\mathbf{G}^p$, where $\mathbf{S}^v \subseteq \mathbf{S}^v$, $\mathbf{E}^v \subseteq \mathbf{E}^p$. Since $\mathbf{G}^v$ include the nodes with correct labels, it can be used to constrain the segmentation of the current frame $\mathbf{I}$. We implant $\mathbf{G}^v$ into the graph $\mathbf{G} = (\mathbf{S}, \mathbf{E})$ of the current frame $\mathbf{I}$, resulting in $\mathbf{G} = (\mathbf{G}^v, \mathbf{G}^x)$, where $\mathbf{G}^x = (\mathbf{S}^x, \mathbf{E}^x)$ is the set of unlabeled nodes, $\mathbf{S}^v \cap \mathbf{S}^x = \emptyset$, $\mathbf{S}^v \cup \mathbf{S}^x = \mathbf{S}$. Hence, the spatio-temporal grouping of the current frame $\mathbf{I}$ is equivalent to an optimal grouping of $\mathbf{G}^x$ subject to a labeled $\mathbf{G}^v$. The segmentation of a partially labeled image (with sparse labeled seeds) has been addressed in [1, 12] as an optimal cut on a partially labeled graph using min-cut/max-flow or random walker. In a two-label case, it is possible to find a global optimum because the energy function is convex. In comparison with a spatial grouping of $\mathbf{G}^x$ subject to $\mathbf{G}^v$ in video segmentation, $\mathbf{L}^x \subset \mathbf{G}^x$ may differ from $\mathbf{L}^v \subset \mathbf{G}^v$ due to dynamic scene changes. The existing labels in $\mathbf{L}^{\mathbf{x}}$ may not fully appear in $\mathbf{L}^v$, and $\mathbf{L}^{\mathbf{x}}$ can also contain some new labels. This fundamental difference makes the spatial grouping of $\mathbf{G}^x$ even more complicated. In this paper, we solve it by a pairwise aggregation of subgraphs based on color heterogeneity, edge strength and shape compactness.

3.1. Propagation

The graph $\mathbf{G}^p$ is reconstructed in $\mathbf{I}$ from the labeled graph $\mathbf{G}^-$ based on the geometric transformation relating the two frames. Without loss of generality, we approximate the inter-frame global motion by an affine transformation $\mathbf{A}$. Then, $\mathbf{I}^-$ is warped to $\mathbf{I}$ by

$$\mathbf{A}\mathbf{I}^- = \mathbf{I} \tag{5}$$

The above linear system can be solved by using $N \geq 3$ corresponding pairs between $\mathbf{I}^-$ and $\mathbf{I}$. We employ the SIFT algorithm in [7] to find and correspond scale-invariant features between $\mathbf{I}^-$ and $\mathbf{I}$. In fact, corresponding pairs undergoing independent motions can cause errors in estimating $\mathbf{A}$. For a robust solution, we use the *RANSAC* algorithm to reject the outliers and minimize the transformation error. With the transformation $\mathbf{A}$, $\mathbf{G}^p$ is constructed by projecting all labeled nodes in $\mathbf{S}^-$ into $\mathbf{I}$. The node edges $\mathbf{E}^p$ are reconnected in the topology of $\mathbf{I}$. It is noteworthy that some nodes in $\mathbf{S}^p$ may be not fully 4-connected due to the geometric transformation.

3.2. Validation

The graph propagation $\mathbf{G}^-$ to $\mathbf{G}^p$ relies only on the estimation of inter-frame global motion. Due to errors introduced by the affine approximation and independent motions, some nodes in $\mathbf{S}^p$ are wrongly labeled. We validate the node labels based on color similarity. For each label l_m^- in $\mathbf{G}^-$, color variances $\sigma_m^-(r)$, $\sigma_m^-(g)$, $\sigma_m^-(b)$ are calculated for all nodes with label l_m^-. Given a node s_n^p in $\mathbf{G}^p$ and its corresponding node s_n^- in $\mathbf{G}^-$, s_n^p is properly labeled if and only if these conditions are satisfied:

$$\begin{aligned} |s_n^p(r) - s_n^-(r)| &\leq 3\sigma_{l^-(s_n)}^-(r) \\ |s_n^p(g) - s_n^-(g)| &\leq 3\sigma_{l^-(s_n)}^-(g) \\ |s_n^p(b) - s_n^-(b)| &\leq 3\sigma_{l^-(s_n)}^-(b) \end{aligned} \tag{6}$$

Image noise often causes random color variations between two corresponding pixels. Instead of performing on a stand-alone node (6) , we validate a node label by its local neighbors (e.g. 3×3 neighbors). With all properly labeled nodes in $\mathbf{S}^p$, a new graph $\mathbf{G}^v = (\mathbf{S}^v, \mathbf{E}^v, \mathbf{L}^-)$ is formed to retain correct labeling information from $\mathbf{G}^-$.

3.3. Independent Motions

The geometric relation in (5) can only recover the inter-frame global motion. It fails to compensate for pixel displacements due to independent motions. These independent motions can be identified by graph validation. Assume that one segmented region r experiences an inter-frame independent motion. Let $\mathbf{g}_r^- = (\mathbf{s}_r^-, \mathbf{e}_r^-, l_r^-)$ be the subgraph of r in $\mathbf{I}^-$. When $\mathbf{g}_r^-$ is propagated to $\mathbf{g}_r^p$ by $\mathbf{A}$, $\mathbf{s}_r^-$ is wrongly located in $\mathbf{I}$. As a result, l_r^- fails the validation check. Let the subgraph $\mathbf{g}_x^p = (\mathbf{s}_x^p, \mathbf{e}_x^p, l_x^-)$ represent the actual location of r in $\mathbf{I}$. Consequently, l_x^- is also invalidated due to color dissimilarity.

Fig. 2 shows an example in which a pingpong ball moves independently with respect to the inter-frame global motion. Given the subgraph of pingpong ball $\mathbf{g}_1^-$ in $\mathbf{I}^-$, graph propagation provides an improper location $\mathbf{g}_1^p$ for it in $\mathbf{I}$. The label of subgraph $\mathbf{g}_1^p$ is invalidated, because the ball colors in $\mathbf{I}^-$ are different from the wall colors in $\mathbf{I}$. The proper location of the ball is indicated by the subgraph $\mathbf{g}_2^p$. The pre-propagated $\mathbf{g}_2^-$ in $\mathbf{I}^-$ is inside the wall. Therefore, the label of subgraph $\mathbf{g}_2^p$ is also invalidated. For a segmented region with independent inter-frame motion, we have the following remark regarding graph propagation and validation,

Remark 1 The independent motion of a segmented region causes the label of its propagated subgraph to be invalidated in $\mathbf{I}$. The label of subgraph at its actual location in $\mathbf{I}$ is also invalidated.

In the case of whole region displacement, such as the illustration in Fig. 2, we match the two invalidated subgraphs based on shape similarity and exchange their labels. Otherwise, if there is an overlap between the projected region and the actual region in $\mathbf{I}$, reassignment of region labels will be handled by graph aggregation. This will be elaborated on in Section 3.4. Following this rectification, we implant $\mathbf{G}^v$ into the graph $\mathbf{G} = (\mathbf{S}, \mathbf{E})$ of $\mathbf{I}$. Then, $\mathbf{G}$ can be classified into labeled and unlabeled sets, $\mathbf{G} = (\mathbf{G}^v, \mathbf{G}^x)$. $\mathbf{G}^x$ is the set of unlabeled nodes, it includes two sets $\mathbf{G}^x = (\mathbf{G}^o, \mathbf{G}^n)$, where $\mathbf{G}^o$ is the set of propagated nodes, and $\mathbf{G}^n$ is set of nodes that newly appear in $\mathbf{I}$ if new objects appear.

3.4. Aggregation

The aggregation of subgraphs performs a spatial grouping for all unlabeled nodes in $\mathbf{G}^x$ based on $\mathbf{G}^v$. The challenge here is that some new groups may be formed in $\mathbf{G}^x$. Instead of using a seeded segmentation as in [1, 12], we conduct a pairwise subgraph grouping on $\mathbf{G}^x$, which is similar to [3], but with different grouping criteria. Prior to the aggregation of subgraphs in $\mathbf{G}^x$, we group the unlabeled nodes in $\mathbf{S}^x$ into small subgraphs by a low-level color clustering (*Mean Shift* [2]). It is conducted to accelerate the grouping of $\mathbf{G}^x$ and to initialize reasonable scales for the subsequent groupings. In a pairwise subgraph grouping, each subgraph $\mathbf{g}^\mathbf{x}$ corresponds to an intermediate group in $\mathbf{I}$. Grouping criteria include edge, color, and shape measures.

3.4.1 Edge

The color gradient between two pixels is characterized by the weight associated with the edge connecting their respective nodes. Let e_{ij} be the edge of two neighboring nodes s_i and s_j. The edge weight is defined by

Definition 1 The edge weight $w(e_{ij})$ between two neighboring nodes s_i and s_j is the norm of L*u*v* color difference between two pixels connected by the edge

$$w(e_{ij}) = \sqrt{(l_i - l_j)^2 + (u_i - u_j)^2 + (v_i - v_j)^2} \quad (7)$$

A strong edge connecting two subgraphs discourages the grouping of the said subgraphs. In [3], the grouping predicate checks if the minimum edge weight connecting a pair of subgraphs is large relative to the internal difference within at least one of the subgraphs. The internal difference is defined as the largest edge weight of the minimum spanning tree, which tries to find a maximum gradient from a low gradient path. This measure is very sensitive to image noises. Given a subgraph $\mathbf{g}_i = (\mathbf{s}_i, \mathbf{e}_i)$, we assume $\mathbf{e}_i^B$ to be the edges crossing the region boundary, $\mathbf{e}_i^B \subset \mathbf{e}_i$. Let $w_B(\mathbf{e}_i^B)$ be the strength of the boundary of subgraph $\mathbf{g}_i$, which is given by

Definition 2 The strength of the boundary of a subgraph $\mathbf{g}_i$ is the mean of all edge weights in $\mathbf{e}_i^B$.

In the case of two neighboring subgraphs $\mathbf{g}_i$ and $\mathbf{g}_j$, we let $\mathbf{e}^J = \mathbf{e}_i \cap \mathbf{e}_j$ be the edges connecting boundary nodes between $\mathbf{g}_i$ and $\mathbf{g}_j$ (we call this set of edges the "joint") and let $w_J(\mathbf{e}^J)$ be the strength of this joint $\mathbf{e}^J$. Then, $w_J(\mathbf{e}^J)$ is estimated by

Definition 3 The strength of the joint between $\mathbf{g}_i$ and $\mathbf{g}_j$ is the mean of all edge weights in the set $\mathbf{e}^J$.

In fact, we prefer the weak edges in $\mathbf{e}^J$ when merging $\mathbf{g}_i$ and $\mathbf{g}_j$ into $\mathbf{g}_k$, i.e., $\mathbf{g}_k = \mathbf{g}_i \cup \mathbf{g}_j$ which means a smaller $w_J(\mathbf{e}^J)$ than $w_B(\mathbf{e}_k^B)$. Therefore, we can formulate the cost of merging $\mathbf{g}_i$ and $\mathbf{g}_j$ as follows

$$C_E(\mathbf{g}_i, \mathbf{g}_j) = \begin{cases} 1 & \text{if } w_J(\mathbf{e}^J) \geq w_B(\mathbf{e}_k^B) \\ \frac{w_J(\mathbf{e}^J)}{w_B(\mathbf{e}_k^B)} & \text{otherwise} \end{cases} \quad (8)$$

3.4.2 Color

Color heterogeneity of a subgraph $\mathbf{g}_i$ is computed as the sum of color variances for all color channels, i.e. $C_H(\mathbf{g}_i) = \sigma_L(\mathbf{g}_i) + \sigma_u(\mathbf{g}_i) + \sigma_v(\mathbf{g}_i)$. Given two neighboring subgraphs $\mathbf{g}_i$ and $\mathbf{g}_j$, the merging cost in terms of color heterogeneity is computed by

$$C_H(\mathbf{g}_i, \mathbf{g}_j) = \begin{cases} 1 & \text{if } C_H(\mathbf{g}_k) \geq avg(i,j) \\ \frac{C_H(\mathbf{g}_k)}{avg(i,j)} & \text{otherwise} \end{cases} \quad (9)$$

where $avg(i,j) = (C_H(\mathbf{g}_i) + C_H(\mathbf{g}_j))/2,\ \mathbf{g}_k = \mathbf{g}_i \cup \mathbf{g}_j$.

3.4.3 Shape

The merging of two subgraphs $\mathbf{g}_i$ and $\mathbf{g}_j$ into $\mathbf{g}_k$ results in a more compact representation of subgraph $\mathbf{g}_k$. The compactness of a subgraph $\mathbf{g}_i$ is used as a generic shape measure. It is defined as $C_S(\mathbf{g}_i) = 4A(\mathbf{g}_i)/L(\mathbf{g}_i)^2$, where $A(\mathbf{g}_i)$ is the area of $\mathbf{g}_i$, and $L(\mathbf{g}_i)$ is the perimeter of $\mathbf{g}_i$. When $\mathbf{g}_i$ is circle, $C_S(\mathbf{g}_i) = 1$. If $\mathbf{g}_i$ is infinitely long and narrow, $C_S(\mathbf{g}_i) = 0$. Given two neighboring subgraphs $\mathbf{g}_i$ and $\mathbf{g}_j$, the cost of merging $\mathbf{g}_i$ and $\mathbf{g}_j$ in terms of shape compactness is given by

$$C_S(\mathbf{g}_i, \mathbf{g}_j) = 1 - \frac{4A(\mathbf{g}_k)}{L(\mathbf{g}_k)^2} \quad (10)$$

3.4.4 Cost

The total cost of merging two subgraphs $\mathbf{g}_i$ and $\mathbf{g}_j$ is the weighted sum of the following measures: color heterogeneity, edge strength and shape compactness. This is given by

$$C(\mathbf{g}_i, \mathbf{g}_j) = k_E C_E(\mathbf{g}_i, \mathbf{g}_j) + k_H C_H(\mathbf{g}_i, \mathbf{g}_j) + k_S C_S(\mathbf{g}_i, \mathbf{g}_j) \quad (11)$$

where k_E, k_H and k_S are weighting factors for edge, color and compactness costs respectively.

A pairwise subgraph aggregation is conducted by searching the best fitting pair of adjacent subgraphs by the rule of mutual best fitting. Let C_{max} be the maximum merging cost. For the subgraph $\mathbf{g}_i$, a neighboring subgraph $\mathbf{g}_j$ is regarded as a merging candidate iff,

$$C(\mathbf{g}_i, \mathbf{g}_j) <= C_{max} \tag{12}$$

For the subgraph $\mathbf{g}_i$, we treat $\mathbf{g}_j$ as the best fitting subgraph among all neighbors of $\mathbf{g}_i$ if a lowest merging cost exists between $\mathbf{g}_i$ and $\mathbf{g}_j$. According to the rule of mutual best fitting, the subgraph $\mathbf{g}_i$ has to be the best fitting neighbor of $\mathbf{g}_j$ as well. Note that the neighboring subgraphs can be labeled or unlabeled. The algorithm of a pairwise subgraph aggregation is summarized in Algorithm 1.

Algorithm 1 A pairwise subgraph aggregation

Require: $\mathbf{G}^x$, $\mathbf{G}^v$

1: Start with the initial subgraphs in $\mathbf{G}^x$.
2: Construct the adjacency relations of these subgraphs.
3: Calculate the merging cost for all adjacent pairs of subgraphs using (11).
4: **repeat**
5: Search the adjacent subgraphs that satisfy (12)
6: Find the best pair of subgraphs $(\mathbf{g}_i, \mathbf{g}_j)$ with the minimum merging cost.
7: Merge $\mathbf{g}_i$ and $\mathbf{g}_j$ into a new subgraph $\mathbf{g}_k = (\mathbf{g}_i, \mathbf{g}_j)$
8: Update the adjacency relations of $\mathbf{g}_k$.
9: Extend the label to $\mathbf{g}_k$ if $\mathbf{g}_i$ or $\mathbf{g}_j$ is labeled.
10: **until** No more pairs of subgraphs satisfy (12).
11: Assign the new labels to the unlabeled subgraphs.

The computational complexity is determined by the number of initial subgraph N_s, and the number of adjacent segments per subgraph N_a. In step 2, the computation to construct the adjacency relations is at most $O(N_s N_a \log(N_s N_a))$. The initial number of possible subgraph pairs $N_s N_a$ is gradually decreased as step 2 proceeds. To update the adjacency relations of one subgraph in step 8, the computation is at most $O(\log(N_s N_a))$. The maximum number of updated subgraphs is $2N_a$. Steps 4-10 are repeated for at most N_s times. The computational complexity is $O(N_s 2N_a \log(N_s N_a))$.

During the subgraph grouping, some small subgraphs (with irregular shapes) are quite resistent to the merging. In this case, we perform a simple smoothing on them using the nearest neighbor based on color similarity after the above grouping procedure.

3.5. Algorithm

The algorithm of the proposed spatio-temporal segmentation is summarized as follows

Algorithm 2 Spatio-temporal segmentation using a preceding graph

Require: $\mathbf{I}^-$, $\mathbf{I}$, $\mathbf{G}^-$

1: Estimate the affine transformation $\mathbf{A}$ using SIFT.
2: Propagate $\mathbf{G}^-$ to $\mathbf{G}^p$ based on (5).
3: Validate the labels $\mathbf{L}^p$ in $\mathbf{G}^p$ using (6). Construct the graph $\mathbf{G}^v$ that contains all trustable labels propagated from $\mathbf{G}^p$.
4: Correct labels of independent moving regions.
5: Implant $\mathbf{G}^v$ into $\mathbf{G}$. Group unlabeled nodes $\mathbf{S}^x$ into small regions using *Mean Shift*
6: Perform subgraph aggregation for unlabeled subgraphs using algorithm 1.
7: Return the labeled $\mathbf{G}$.

4. Experiments and Discussion

To test the validity of the proposed algorithm, we have applied it on several typical test sequences, namely, the "Table Tennis", the "Coast Guard" and the "Jumping Girl" sequences. We will focus our discussion on the first two sequences. Initialization of first frame of the video sequences was done by applying *Mean Shift* segmenter. Minimum user intervention is required in tuning the parameters for the segmenter. The over-segmentation obtained was merged according to region similarity measures to form initialized segmentation for every sequence. We present results for video sequences in which different challenges arise. In the "Table Tennis" sequence, there is high temporal activity due to rapidly-changing independent motion. In the second sequence, the small sizes of independent moving objects and their blurry edges make it difficult to contrast against the background. Default parameters used in the total cost formula (11) are: $k_E = 0.3$, $k_H = 0.6$ and $k_S = 0.2$.

4.1. Overall Segmentation Evaluation

To objectively evaluate the video segmentation quality, we refer to [10] for quality evaluation measures. Experimental results are compared by overlaying the segmentation with their manually segmented ground-truths. Fig.3 shows the overall segmentation accuracy for frames $1-30$ of the "Table Tennis" sequence and that for frames $10-35$ of the "Coast Guard" sequence. This overall segmentation accuracy is defined as,

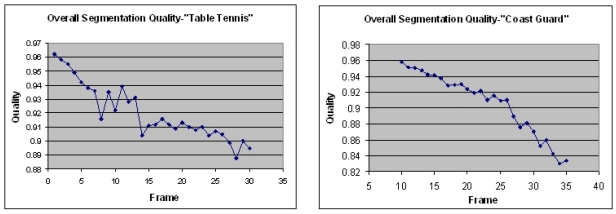
$$Accu(\mathbf{S}) = \sum_{\mathbf{s}=\mathbf{s}_1}^{\mathbf{s}_N} \frac{N_{accu}(\mathbf{s})}{N_{total}(\mathbf{s})} \tag{13}$$

where $N_{accu}(\mathbf{s})$ is the number of correctly labeled pixels in $\mathbf{s}$, and $N_{total}(\mathbf{s})$ is the number of pixels in $\mathbf{s}$.

Fig.6 and Fig.7 show selected segmentation results for sections of both sequences. The overall segmentation quality corresponding to the 2 sequences are presented in Fig. 3.

Table 1. Average percentage of propagated, validated and new pixels for (1) frames 1−30 of "Table Tennis (TT)" Sequence and (2) frames 10−35 of "Coast Guard (CG)" Sequence.

Seq.	Class	Propagated(%)	Validated(%)	New(%)
TT	Table	96.20	85.10	0.20
	Ball	98.35	0.29	0
	Hand	97.50	12.18	6.57
CG	Boat	97.50	87.10	0.11
	Water	98.35	84.20	5.22
	Land	97.21	95.50	5.43



(a) (b)

Figure 3. Overall segmentation quality: (a) Overall quality for frames 1−30 of the "Table Tennis" sequence. (b) Overall quality for frames 10−35 of the "Coast Guard" sequence

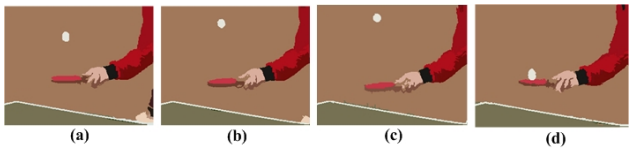


Figure 4. (a)−(d) Segmentation results of frames 1, 5, 9 and 12 in the "Table Tennis" sequence. The pingpong ball and human hand are segmented as independent moving objects. Note that pingpong ball is correctly associated despite no temporal overlapping after propagation.

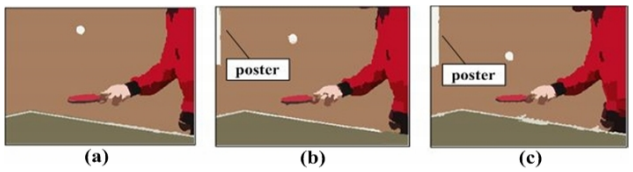


Figure 5. (a)−(c) Segmentation results for frame 34, 37 and 39 of the "Table Tennis" sequence. The poster on the wall is successfully detected and segmented as a newly appeared object.

The overall accuracy for the "Table Tennis" sequence drops from an initial value of 96.35% to the lowest value of 88.9% as the temporal section approaches its end. Similar results are observed for the "Coast Guard" sequence, with a maximum drop of 13%. The decline in overall accuracy is due to accumulation of propagation error. The results shown also reflect a tolerance limit for acceptable deterioration. Temporal graph validation verifies the predicted pixel labels after propagation, but it does not guarantee an error-free graph aggregation. Some error residual will still be carried over to subsequent frames. Empirically, it is found that to limit the temporal error propagation to within 10%, the maximum propagation time span allowed is about 20 frames. Furthermore, to highlight the profitable exploitation of temporal redundancy in video segmentation, Table 1 shows the average percentage of propagated, validated and newly appeared pixels for both video sections. On comparing the percentage of validated pixel labels for both sections, we can see that the percentage of validated labels is more than 84.20% when there is little or no independent motions (Table 1, seq. "CG"). Selected segmentation results for the "Jumping Girl" sequence is shown in Fig. 8.

4.2. Independent Motion

As discussed in Section 3.3, the proposed algorithm is designed to handle independent motions. Fig. 4 illustrates a case of fast independent motion. The video section we have tested on (frames 1−30 of "Table Tennis" sequence) contains fast independent motions. The pingpong ball bounces up and down and the human arm, an articulated model, swings back and forth. Traditional approaches based on motion parameters estimation suffer from their inability to handle fast-moving objects, while the proposed algorithm is able to track both the pingpong ball and the arm accurately.

4.3. Newly Appeared Objects

Newly appeared objects are detected during graph aggregation as "unmerged" regions. Fig. 5 shows a case where a poster hanging on the wall enters the scene. The proposed algorithm is able to detect this newly appeared object. Despite its color similarity to the pingpong ball and the table edge, proximity constraint (only neighboring subgraphs are merged during pair-wise subgraph grouping) is still able to identify this object as a new comer.

5. Conclusion

In this paper, we have presented an efficient algorithm to exploit the inter-frame correlation to propagate trustable grouping from the previous frame to the current. A preceding graph is built and labeled for the previous frame. It is temporally propagated to the current frame, validated by the similarity measures, and spatially aggregated for the final grouping. In doing so, we retain maximally the propagated segmentation results and hence lessen the computational burden of re-segmenting every frame. Experimental results show that the proposed algorithm is able to handle fast independent motion and appearance of new objects through graph validation and aggregation processes. For future work, a more robust treatment of subgraph validation

(a) (b) (c) (d) (e)

Figure 6. Selected segmentation results for the frames 1−30 in the "Table Tennis" sequence: (a) Initialized segmentation for frame 1. (b)−(e) Segmentation results for frames 3, 7, 13 and 25 respectively.

(a) (b) (c) (d) (e)

Figure 7. Selected segmentation results for the frames 10−35 in the "Coast Guard" sequence: (a) Initialized segmentation for frame 10. (b)−(e) Segmentation results for frames 13, 19, 27 and 33 respectively.

Figure 8. Selected segmentation results for the frames 1−30 in the "Jumping Girl" sequence: (a)Initialized segmentation for frame 1. (b)−(d)Segmentation results for frames 5, 15 and 30 respectively.

such as incorporating multiple low-level features would be beneficial to the segmentation.

Acknowledgment

This work has been supported by research grant DSOCL06064.

References

[1] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[2] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[3] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[4] J. Goldberger and H. Greenspan. Context-based segmentation of image sequences. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(3):463–468, 2006.

[5] B. Li and R. Chellappa. Face verification through tracking facial features. *Journal of the Optical Society of America : A*, 18:2969–2981, 2001.

[6] S. Z. Li. Markov random field modeling in image analysis, 2nd edition. 2001.

[7] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[8] R. Megret and D. DeMenthon. A survey of spatio-temporal grouping techniques, 1994. Technical report: LAMP-TR-094/CS-TR-4403, University of Maryland, College Park.

[9] I. Patras, E. A. Hendriks, and R. L. Lagendijk. Video segmentation by map labeling of watershed segments. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(3):326–332, 2001.

[10] P.L.Correia and F.Pereira. Objective evaluation of video segmentation quality. *IEEE Trans. Image Processing*, 12(2):186–200, 2003.

[11] E. Sharon, A. Brandt, and R. Basri. Fast multiscale image segmentation. *Proc. IEEE International Conference on Computer Vision*, pages 70–77, 1999.

[12] A. K. Sinop and L. Grady. A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. pages 290–294, 2007.

[13] Y. Wang, K. F. Loe, T. Tan, and J. K. Wu. H.264 and mpeg-4 video compression: Video coding for next-generation multimedia. *IEEE Trans. Image Processing*, 14(7):937–947, 2005.