

Chapter 2

Image Segmentation with Eigen-Subspace Projections

Jar-Ferr Yang and Shu-Sheng Hao

Abstract In this chapter, object segmentation algorithms dependent on the characteristics of eigen-structure are proposed. The eigen-subspaces are obtained from eigen-decomposition of the covariance matrix, which is computed from the selected color samples. Hence, the color space can be transformed into the signal subspace and its orthogonal noise subspaces. After statistical analysis of eigen-structure of target color samples, the color eigen-structure segmentation algorithms are then designed to extract the desired objects, which are close to the color samples. The principal component transformation (PCT) techniques, which only use the signal subspace can be treated as a subset of color eigenspace algorithms. The eigenspaces discriminated as signal and noise subspaces related to original color samples should be effectively utilized. The adaptive eigen-subspace segmentation (AESS) algorithm, which can save the computation of eigen-decomposition, is applied to adaptively adjust the eigen-subspaces. Finally, the Eigen-based fuzzy C-means (FCM) clustering algorithm has been proposed to effective segment color object. By jointly consideration of signal and noise subspace projections of desired colors, the separate eigen-based FCM (SEFCM) and coupled eigen-based FCM (CEFCM) are used to achieve effective color object segmentation. With these proposed algorithms, the color objects can be successfully extracted by using eigen-subspace projections.

2.1 Overview

Image segmentation has been treated as a key technology in many smart image and video related applications. To achieve effective coding, for example, the image segmentation is the most important kernel in construction of the MPEG-4 video object plane (VOP) [1, 2]. The four major features, including luminance, motion, color,

J.-F. Yang (✉)

Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan

e-mail: jfyang@ee.ncku.edu.tw

and depth information, have been used as indexes for segmentation of the scenes. If we can separate the video objects and encode them in the video bitstreams, we can achieve the goals such as content scalability, sprite construction, and depth map estimation [3]. Recently, there are many segmentation researches proposed by using motion [4, 5], edge [6, 7], shape [8], and textual [9, 10] information. In this chapter, we introduce the object segmentation algorithms by only using the characteristics of eigenstructure of the color space. After statistical analysis of eigen-structure of the color samples, the color eigen-structure segmentation algorithms, which consider characteristics of signal and noise subspaces, are suggested. By using color information only, simulations show that the proposed algorithm can successfully detect the desired objects from standard video test sequences. In Sect. 2.2, the object segmentation algorithm based on the color eigen-structure characteristics will be stated. In Sect. 2.3, color object segmentation using adaptive eigen-subspaces will be discussed. In Sect. 2.4, color object segmentation using fuzzy C-means with eigen-subspace projection will be described. Conclusions will be stated in Sect. 2.5.

2.2 An Object Segmentation Algorithm Based on Color Eigen-Structure Characterizations

As mentioned in Sect. 2.1, the principal component transformation (PCT) for image segmentation has been proposed [11–13]. The PCT essentially exhibits a color transformation to the signal subspace only. In this section, we propose a color eigen-structure algorithm to efficiently and effectively retrieve the desired objects. In Sect. 2.2.1, the theory of PCT will be introduced. In Sect. 2.2.2, we further analyze the properties of eigen-subspaces. In Sect. 2.2.3, we adopt the statistical analysis of the eigen-structure to design a color eigen-structure segmentation algorithm. The detailed procedures of the algorithm are also described. In Sect. 2.2.4, simulation results will be shown to verify the above theoretical development.

2.2.1 Principal Component Transformation (PCT)

The principal component transform (PCT) [17–19] is also called discrete Karhunen-Loève (KL) expansion. The KL transformation achieves optimal energy compaction and independent properties, which are commonly used for data compression. For the purpose of color object segmentation, the PCT could help to identify the most likely component. The proposed algorithms can also apply to other color coordinates, for example, YUV or YC_bC_r as well. Without losing the generality, we choose RGB components to form the covariance matrix related to the selected color samples. First, using mouse clicks on the desired object to choose a few desired color samples. The k th sample in the RGB color vector is given by

$$\mathbf{s}_k = [r_k \ g_k \ b_k]^T, \quad (2.1)$$

where r_k, g_k , and b_k are red, green, and blue levels of the k th sample in each color plane. In (2.1), the superscript T denotes the transpose of the argument vector. Given M color samples, we can compute the covariance matrix, \mathbf{R}_s as

$$\mathbf{R}_s = \frac{1}{M} \sum_{k=1}^M \mathbf{s}_k \mathbf{s}_k^T. \quad (2.2)$$

Applying the eigen-decomposition procedure on the matrix \mathbf{R}_s , we can obtain three eigenvectors \mathbf{w}_1 , \mathbf{w}_2 , and \mathbf{w}_3 . The eigenvectors are corresponding to the eigenvalues λ_1 , λ_2 , and λ_3 , which are arranged in the descending order as

$$\lambda_1 \geq \lambda_2 \geq \lambda_3. \quad (2.3)$$

The covariance matrix \mathbf{R}_s can be expressed by

$$\mathbf{R}_s = \sum_{i=1}^3 \lambda_i \mathbf{w}_i \mathbf{w}_i^T. \quad (2.4)$$

The first principal component \mathbf{w}_1 corresponding to the largest eigenvalue becomes the best representation of the desired data samples. If any unknown color samples possess large projections along \mathbf{w}_1 , we may treat those samples to have a higher possibility with the same classification in color as the selected samples. In order to obtain more satisfactory results, we should jointly consider \mathbf{w}_1 , \mathbf{w}_2 , and \mathbf{w}_3 projections. We can divide the projected space into two kinds of subspaces, i.e. signal subspace and noise subspaces. The signal subspace is formed by the eigenvector \mathbf{w}_1 associated with the largest eigenvalue λ_1 while the noise subspaces are constructed by the eigenvectors \mathbf{w}_2 and \mathbf{w}_3 corresponding to λ_2 and λ_3 . It is noted that the eigenvectors, \mathbf{w}_1 , \mathbf{w}_2 , and \mathbf{w}_3 of any covariance matrix are orthonormal vectors. Thus, the signal and noise subspaces are orthogonal with each other.

For most PCT methods [14–16], that used the first principal component for color object extraction, they would face the problem in determination of their thresholds by statistical analyses of eigen-structures [17]. For semiautomatic color object segmentation, the sampled pixels could be obtained from mouse clicks upon the desired color objects. With the computed or prestored eigen-structures, the PCT method can extract the features in some conditions [18–20]. In order to localize the desired object in the image, the adaptive eigen-subspaces method will be used to extract the interesting color object [21]. However, the detection performance of the PCT method will be degraded if the color samples are not properly adopted. To achieve satisfactory segmentation, we should further cooperate with iterative or fuzzy inferences to improve the PCT method [22]. In order to extract meaningful objects in different images, we can collect all desired colors to setup color subspaces for initialization of the fuzzy clustering algorithms.

2.2.2 Color Eigen-Subspaces

Assumed the desired objects exhibit an average color sensation, which previously is expressed as (2.1). It is noted that the derivations can be applied to any other color space. However, we develop the algorithm for the RGB color space only. In order to divide the three-dimension color spaces into noise and signal subspaces, we further assume that the desired objects contain no more than two large-displaced colors in the average sense. In other words, the number of the desired colors is limited to $p = 1$ or 2 . As to the texture or the shadow effect of the desired objects, the variation of colors in the desired objects are modeled as independent noises and expressed by

$$\mathbf{g}_k = \mathbf{s}_k + \mathbf{n}_k = [r_k \ g_k \ b_k]^T + [n_{r,k} \ n_{g,k} \ n_{b,k}]^T, \quad (2.5)$$

where $n_{r,k}$, $n_{g,k}$, and $n_{b,k}$ are the k th sampled color noises, which are assumed to be statistically independent to the desired color vector \mathbf{s}_k and uncorrelated with each other. The covariance matrix \mathbf{R}_s of the sampled color vectors is defined as

$$\mathbf{R}_g = E [\mathbf{g}_k \mathbf{g}_k^T]. \quad (2.6)$$

Since the number of average color vectors, p is limited under two, i.e., $p = 1$ or 2 , the noise free color covariance matrix can be expressed by p principal components as

$$\mathbf{R}_g = \sum_{i=1}^p \lambda_i \mathbf{v}_i \mathbf{v}_i^T, \quad (2.7)$$

where λ_i represents the i th eigenvalue of \mathbf{R}_g and \mathbf{v}_i denotes its corresponding eigenvector. The span of $\mathbf{s}_i, i = 1, \dots, p$ is equal to the span of $\mathbf{v}_i, i = 1, \dots, p$, which is called the signal subspace. Due to the independent assumption of sample noises, the covariance matrix of the sample noises can be modeled as

$$\mathbf{R}_n = \sigma_n^2 \mathbf{I}. \quad (2.8)$$

The covariance matrix of sampled color vectors composed of both signal and noise components can be expressed by

$$\mathbf{R}_g = \mathbf{R}_s + \mathbf{R}_n = \sum_{i=1}^p \lambda_i \mathbf{v}_i \mathbf{v}_i^T + \sigma_n^2 \mathbf{I} = \sum_{i=1}^p (\lambda_i + \sigma_n^2) \mathbf{v}_i \mathbf{v}_i^T + \sum_{i=p+1}^3 \sigma_n^2 \mathbf{v}_i \mathbf{v}_i^T. \quad (2.9)$$

It is noted that the random noises in the average sense do not change the direction of original signal subspace but add the noise power (variation) σ_n^2 to the true eigenvalues of \mathbf{R}_s . The remaining subspaces in the RGB color coordinate system, which are called the noise subspaces, become the span of $\{\mathbf{v}_i, i = (p+1), \dots, 3\}$. It is obvious that the eigenvectors of a symmetrical matrix are orthogonal to each other. Accordingly, the signal subspace and noise subspace will be orthogonal to each other. For example, if we choose the skin as the desired objects by choosing $p = 1$,

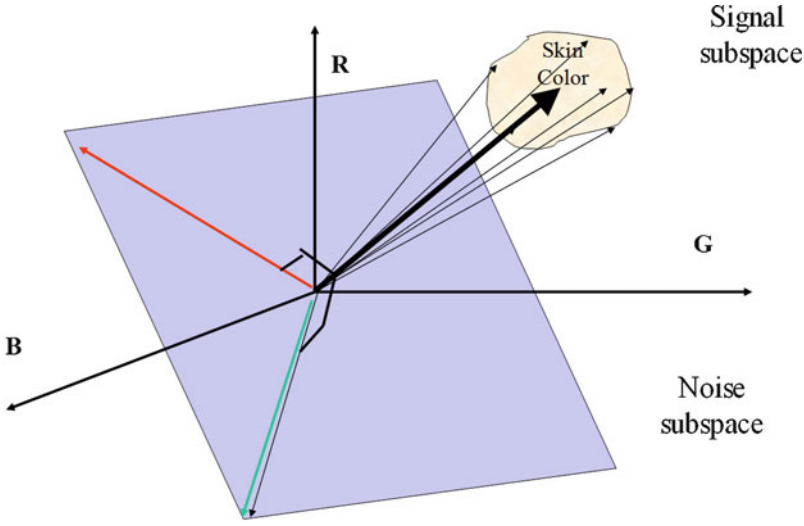
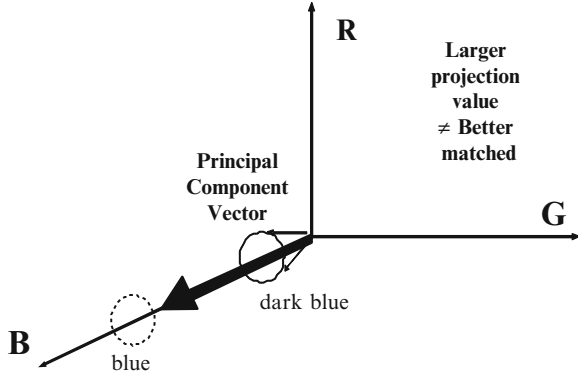


Fig. 2.1 The relationship of signal and noise subspaces

Fig. 2.2 Problems in the first principal component with larger projection (The solid region (dark blue) and dot region (blue) have the same direction)



the signal is the span of $\{\mathbf{v}_1\}$ and the noise subspaces become the span of $\{\mathbf{v}_2, \mathbf{v}_3\}$. Figure 2.1 shows the relationship of signal and noise subspaces for the skin objects.

In order to segment the desired object, the most frequent approaches perform the so-called principal color segmentation [23–27]. First, we can obtain the sampled covariance matrix followed by an eigen-decomposition to obtain \mathbf{v}_1 , the eigenvector, which is corresponding to the largest eigenvalue. With the principal component vector at hand, we then project all the color vectors of image pixels to the principal component vector \mathbf{v}_1 . Finally, the object segmentation can be achieved by choosing the pixels, which have the largest projections with a proper threshold. The threshold method of the principal value technique is widely adopted for many optimization applications. However, there are two problems that arise in use of the first principal color component for color object segmentation. Figure 2.2 shows the fact that the larger the projection, it implies the better match of color. From the viewpoint of

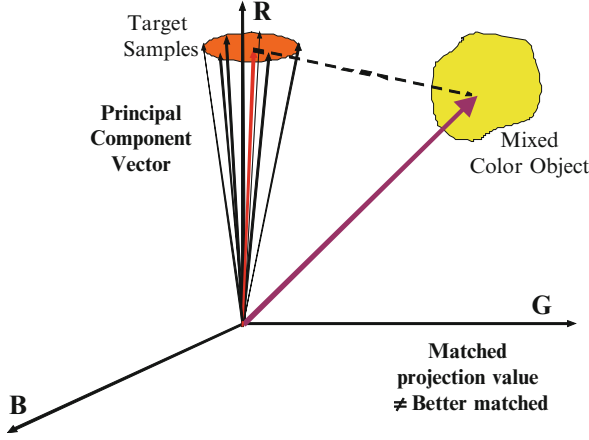


Fig. 2.3 Problem in the first principal component with different hues

color perception, the strong intensity color should be totally different from the low intensity color even if they share the same color space. To classify the dark-blue cloth color, the light-blue sky color, however, will have larger projection. Figure 2.3 shows the case that the mixed colors could have the same color projection although they have different hues. The pure yellow color ($R+G$) and the pure red color will have totally the same projection if the pure red color object is selected. The highly mixture color such as the high intensity white color, usually has larger projection. Hence, we need exploit the complete eigen-structure to develop a color segmentation algorithm to overcome those deficiencies in using the principal component approaches.

2.2.3 Color Eigen-Subspaces Segmentation

For given sampled color vectors \mathbf{g}_k of the target objects, we can obtain the sampled covariance matrix $\hat{\mathbf{R}}_g$ as (2.6). Through the eigen-analysis procedures, we can obtain the eigenvectors $\hat{\mathbf{v}}_1$, $\hat{\mathbf{v}}_2$, and $\hat{\mathbf{v}}_3$ corresponding to the eigenvalues, $\hat{\lambda}_1$, $\hat{\lambda}_2$, and $\hat{\lambda}_3$, which are arranged in the descending order as

$$\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \hat{\lambda}_3. \quad (2.10)$$

Since the covariance matrix $\hat{\mathbf{R}}_g$ can be expressed by

$$\hat{\mathbf{R}}_g = \sum_{i=1}^p (\hat{\lambda}_i + \hat{\sigma}_n^2) \hat{\mathbf{v}}_i \hat{\mathbf{v}}_i^T + \hat{\sigma}_n^2 \sum_{i=p+1}^3 \hat{\mathbf{v}}_i \hat{\mathbf{v}}_i^T. \quad (2.11)$$

The least eigenvalue of the sampled covariance matrix can be used as the estimator of the noise power as

$$\hat{\sigma}_n^2 \approx \hat{\lambda}_3, \quad (2.12)$$

and the corresponding eigenvector is estimated by $\mathbf{v}_3 \approx \hat{\mathbf{v}}_3$. The estimated principal eigenvalues are given by

$$\lambda_1 \approx \hat{\lambda}_1 - \hat{\lambda}_3, \quad (2.13)$$

and

$$\lambda_2 \approx \hat{\lambda}_2 - \hat{\lambda}_3. \quad (2.14)$$

While the estimated eigenvectors are $\mathbf{v}_1 \approx \hat{\mathbf{v}}_1$ and $\mathbf{v}_2 \approx \hat{\mathbf{v}}_2$. Before the development of our segmentation algorithm, we should further analyze the statistical properties of the above estimators such that the thresholds of the signal and noise subspaces are reasonably designed.

2.2.3.1 Statistical Analysis of Eigen-Structures

The covariance matrix of the interested spatial samples is given by \mathbf{g}_k as in (2.5). The covariance matrix \mathbf{R}_g and its estimated matrix $\hat{\mathbf{R}}_g$ are respectively defined in (2.9) and (2.11). Based on $\hat{\mathbf{R}}_g$, we can obtain the estimated signal and noise subspaces. To explore their expectations and deviations, we should first analyze the asymptotic statistics for the eigenvalues and eigenvectors of the sampled covariance $\hat{\mathbf{R}}_g$ under the Gaussian process assumption [28]. Based on the perturbation formulation, the first- and second-order moments of $\hat{\lambda}_i$ and $\hat{\mathbf{v}}_i$ can be obtained [20, 29]. The eigenvectors of the signal subspace $\hat{\mathbf{v}}_i$ and its associated eigenvalues $\hat{\lambda}_i$ are asymptotically normal with noise subspaces $\hat{\mathbf{v}}_j$ and $\hat{\lambda}_j$, for $i, j = 1, 2, 3, i \neq j$. According to [20], the expectation value of $\hat{\mathbf{v}}_i$ and the covariance of $\hat{\lambda}_i$ can be expressed by (2.15) and (2.16) as follows:

$$E[\hat{\mathbf{v}}_i] \approx \mathbf{v}_i - \frac{1}{2} \sum_{j=1, j \neq i}^3 \frac{\lambda_i \lambda_j}{(\lambda_j - \lambda_i)^2 N} \mathbf{v}_j, \quad (2.15)$$

$$\mathbf{cov}(\hat{\lambda}_i, \hat{\lambda}_j) \approx \frac{\delta_{i,j} \lambda_i^2}{N}, \quad (2.16)$$

where N is the number of sampled pixels and $\delta_{i,j}$ is the Kronecter delta. The estimated values of $\hat{\mathbf{v}}_i$ and $\hat{\lambda}_i$ can be expressed by

$$\hat{\lambda}_i = \lambda_i + \xi_i, \quad (2.17)$$

and

$$\hat{\mathbf{v}}_i = \mathbf{v}_i + \mathbf{s}_i, \quad (2.18)$$

where the error terms, \mathbf{s}_i and ξ_i have the following asymptotic properties [20]:

$$\sigma_{\lambda_i \lambda_j}^2 = E[\xi_i \xi_j] \approx \frac{\lambda_i^2}{N} \delta_{i,j}, \quad (2.19)$$

$$E[\mathbf{s}_i] \approx -\frac{\lambda_i}{2N} \sum_{k=1, k \neq i}^3 \frac{\lambda_k}{(\lambda_i - \lambda_k)^2} \mathbf{v}_i. \quad (2.20)$$

The noise term ξ_i in (2.17), which has been obtained from (2.19) will be used to set the threshold values on three transformed color spaces in the following section.

2.2.3.2 Object Segmentation Algorithm Based on Color Eigen-Structures

In order to classify the input color vector into the signal and the noise subspace, we can project the color vector on the eigenvectors of the sample covariance matrix as

$$y_{i,k} = \mathbf{v}_i^T \cdot \mathbf{g}_k, \quad \text{for } i = 1, 2, 3. \quad (2.21)$$

Now, we should statistically analyze the projection length of $y_{i,k}$ by taking expectation of the power as

$$E[y_{i,k} y_{i,k}^T] = E[\mathbf{v}_i^T \mathbf{g}_k \mathbf{g}_k^T \mathbf{v}_i] = \mathbf{v}_i^T \hat{\mathbf{R}}_g \mathbf{v}_i = \hat{\lambda}_i. \quad (2.22)$$

Thus, the average length of the eigenvector projection should become

$$|y_{i,k}| = |\mathbf{v}_i^T \cdot \mathbf{g}_k| = \sqrt{\hat{\lambda}_i}, \quad \text{for } i = 1, 2, 3. \quad (2.23)$$

For the principal component approach, we can simply detect the color pixel by choosing

$$|y_{1,k}| = |\mathbf{v}_1^T \cdot \mathbf{g}_k| \geq \sqrt{\hat{\lambda}_1}. \quad (2.24)$$

This is the so-called signal subspace projection. Any pixel color vector, which has large enough projection onto the direction of the principal color vector, will be treated as the object pixel for color segmentation. As Fig. 2.2 shown, the brighter color generally has larger projection. As shown in Fig. 2.3, the mixed color could have the same projection as the target color space. The approach of principal component usually preserves the desired color object, however, erroneously includes brighter color and mixed color pixels.

From the statistics analysis obtained in Sect. 2.2.3.1, we should classify the color space by using both signal and noise subspaces by determining the threshold values in the transformed color spaces. First, we should detect the signal space component more precisely. In order to include 97.5% confidence interval of the principal projection, we propose the signal-subspace detection criterion by modifying (2.24) as

$$\sqrt{\hat{\lambda}_1 + k_{1s} \sigma_{\lambda_1}} \geq |y_{1,k}| \geq \sqrt{\hat{\lambda}_1 - k_{1s} \sigma_{\lambda_1}}, \quad (2.25)$$

where k_{1s} is a constant that equals to 3. The deviation σ_{λ_1} of the first principal eigenvalue is given by

$$\sigma_{\lambda_1}^2 = E[\xi_1 \xi_1] \approx \frac{\lambda_1^2}{N}. \quad (2.26)$$

We relax the lower bound by three deviations to include the possible shadow colors and add the upper bound with three deviations to exclude the unwanted brighter colors. Thus, we can eliminate the incorrect luminance pixels as possible. The pixels, which meet the signal subspace criterion stated in (2.25), could be very possible mixed color pixels.

In order to further exclude the mixed color pixels, we should use the noise space criterion to remove the pixel color pixels from the signal subspace pixels, which satisfy the criterion stated in (2.25). The noise subspace criterion can be discussed in two cases: $p = 1$ and $p = 2$. For $p = 1$, the noise subspace now becomes the span of $\{v_1, v_2\}$. We should perform the noise subspace criterion as

$$|y_{i,k}| = |\mathbf{v}_i^T \cdot \mathbf{g}_k| > \sqrt{\hat{\lambda}_i + k_{in} \sigma_{\lambda_i}}, \quad \text{for } i = 2, 3, \quad (2.27)$$

to remove the unwanted pixels, where k_{in} is a constant to specify the confidence interval of the noise. We know that the pixels, whose projections to the noise subspace should be as small as $\sqrt{\hat{\lambda}_i}$ for $i = 2, 3$, are matched with the desired color modal. For any other pixels with mixed colors, their color vectors project onto the noise subspace will be larger than $\sqrt{\hat{\lambda}_i}$ for $i = 2, 3$. Similarly, we can keep the desired pixels once we find their projections to the noise subspace are beyond the limits of $k_{in} \cdot \sqrt{\hat{\lambda}_i \pm k_{2n} \cdot \sigma_{\lambda_i}}$ for $i = 2, 3$. For $p = 1$, we perform the detection of

$$k_{1n} \cdot \sqrt{\hat{\lambda}_i + k_{2n} \cdot \sigma_{\lambda_2}} \geq |y_{2,k}| = |\mathbf{v}_2^T \cdot \mathbf{g}_k| \geq k_{1n} \cdot \sqrt{\hat{\lambda}_i - k_{2n} \cdot \sigma_{\lambda_2}} \quad (2.28)$$

and

$$k_{1n} \cdot \sqrt{\hat{\lambda}_i + k_{2n} \cdot \sigma_{\lambda_3}} \geq |y_{3,k}| = |\mathbf{v}_3^T \cdot \mathbf{g}_k| \geq k_{1n} \cdot \sqrt{\hat{\lambda}_i - k_{2n} \cdot \sigma_{\lambda_3}} \quad (2.29)$$

to remove the unwanted pixels. We set the constants $k_{1n} = 3$ and $k_{2n} = 3$ in our experiment to achieve the best results. In (2.28) and (2.29), σ_{λ_2} and σ_{λ_3} denote the deviation of the second and third eigenvalues respectively given by

$$\sigma_{\lambda_2}^2 = E[\xi_2 \xi_2] \approx \frac{\lambda_2^2}{N}, \quad (2.30)$$

and

$$\sigma_{\lambda_3}^2 = E[\xi_3 \xi_3] \approx \frac{\lambda_3^2}{N}. \quad (2.31)$$

In summary, we utilize the complete projections of both signal and noise subspaces to detect the desired color pixels. The signal-subspace projection helps to classify the desired pixels while the noise-subspace projection provides the information to eliminate the unmatched pixels. Theoretically, the noise plane related to the smallest eigenvalue can highlight the most wanted object after removing

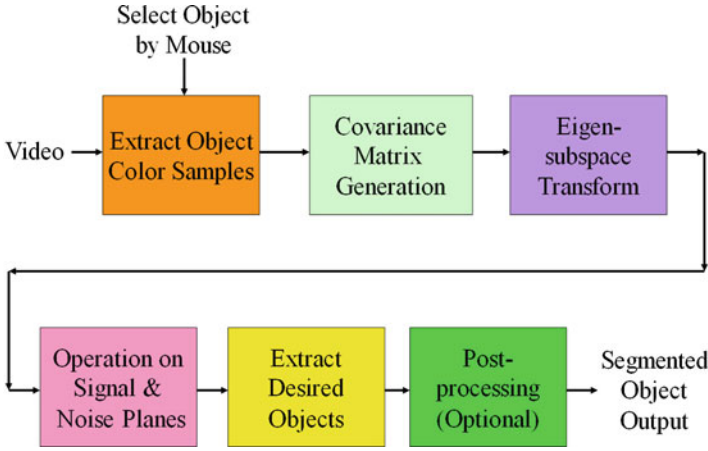


Fig. 2.4 Simulation function block diagram of video object segmentation

noise. After join considering with signal space, the smallest eigenspace is too sensitive to extract the objects that contains too many shadows. These shadows usually are with a similar hue as the desired samples that we should take them as parts of the objects. In order to correct representing the segmented objects, we can take the advantage that the extracted color from the second large noise plane is not so accurate as the smallest one. By inspecting the second noise space related to the second large eigenvalue, we find out that this plane can well describe the silhouette of video object. Hence, the proposed color segmentation using complete eigen-structure should result effective and efficient detection performances in detecting the desired color objects. Figure 2.4 shows the procedures of applying the proposed color eigen-structure segmentation method on some known or instant samples of the desired color object such as skin, hair, or clothes.

2.2.4 Simulation Results

In order to verify the effectiveness of the proposed algorithms, we adopt four standard sequences, Mosaic, Ball, Akiyo, and News, which are shown in Fig. 2.5 in simulations. For each image, we mark with a triangular sign on it to indicate the desired color object we want to extract. Because we have not introduced any spatial and temporal information, all the other similarly colored objects with the same color sensation will also appear in the segmentation results. To clearly exhibit the desired color in the segmented images, we only show the desired color and undesired color objects by bright pixels with gray level = 255 and dark pixels with gray level = 0, respectively. First, we only apply the PCT algorithm to extract the desired color objects. From the signal and noise subspaces, the segmented objects are varied diversely by different threshold setting. By using the threshold determination

Fig. 2.5 Test sequences: (a) Mosaic; (b) Ball; (c) Akiyo; (d) News (The desired color objects are marked with triangles)

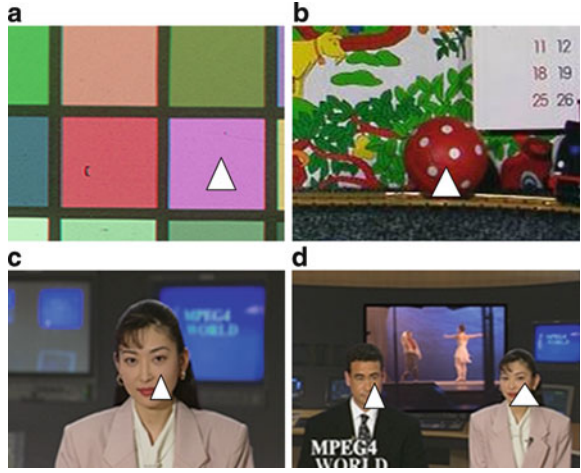
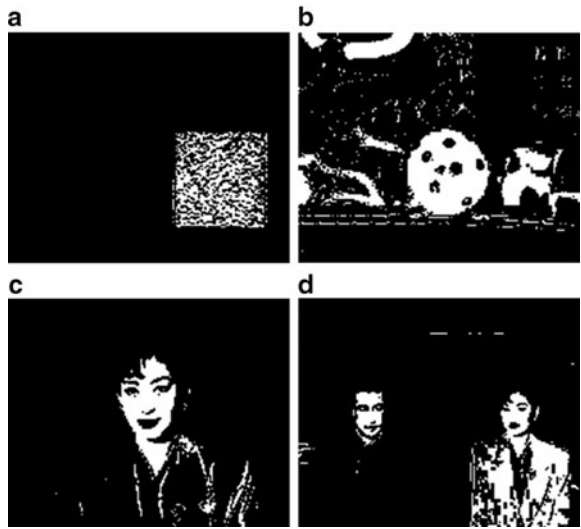


Fig. 2.6 Segmented images obtained by the PCT method using threshold and logical operation: (a) Mosaic; (b) Ball; (c) Akiyo; (d) News sequence



suggested in [17] and applying some logical operations, we can obtain four segmented images as shown in Fig. 2.6. Although the main parts of the desired object are extracted, noise cannot be easily de-correlated from signal.

The test sequences are shown in Fig. 2.5 embedded with different characteristics. The scenes of these four sequences are quite different. Akiyo sequence has a blue static TV screen on her background and with some background color similar to Akiyo's face. Carphone sequence has large head movement and fast scene changes outside the car window. Due to fast head movement, the lightness changes occur on his face very quickly. This lightness effect is usually hard to overcome by other

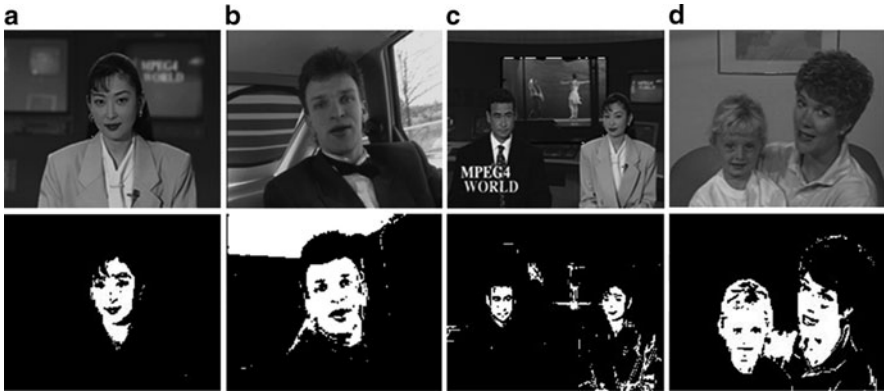


Fig. 2.7 Simulation results of four test sequences: (a) Akiyo; (b) Carphone; (c) News; (d) Mother-and-daughter sequences (*up*: original images; *down*: extracted skin color objects)

segmentation methods. The News sequence has two newscasters that occupy two small regions in the scene. In this sequence, the background is more complex with one static blue screen and a large scene-changing TV screen. Actually, the face skin regions are very small in this sequence. The MD sequence has serious shadow effect on mother's clothes and daughter's head.

While simulation, we extract the desired samples only from the first frame of the sequences to obtain the covariance matrix. The skin color extraction results of the first frame from four test sequences are shown in Fig. 2.7. The images on left column show the original images while the right column exhibits the extracted skin color regions. Simulation results show that those skin regions are successfully extracted by using our algorithm.

Inspecting Fig. 2.7, our algorithm can also identify even the small regions such as eyes and mouth, which are with a different color sensation from the skin. In Fig. 2.7b, we also extract the car's roof because it has similar color as the man's face. Applying the temporal redundancy, of course, we can remove the car's roof by using some motion information. In Fig. 2.7d, the extraction results of Mother's and daughter's faces are influenced by the shadows but the main parts of skin are revealed. Figure 2.8 shows the extraction results of clothes and hair objects. Inspecting Fig. 2.8a,b, we can extract the Akiyo's clothes and hair separately according to different sample color. We can also extract the clothes of News and MD as shown in Fig. 2.8c,d. In order to verify the robustness of our algorithm, we take different frames in Akiyo's sequence with the same transformation matrix obtained from the first frame. From Fig. 2.9, we find that our algorithm can mostly extract the skin regions, in which her different expressions can be also observed.

Generally speaking, it is almost unnecessary to perform any post processing method in our algorithm. Inspecting the simulation results, even the small features such as eyes and mouth can be also indicated. If needed, we can also utilize the temporal information to remove the unwanted static scenes. The signal and noise subspaces' thresholds can be defined according to (2.25), (2.28), and (2.29). In

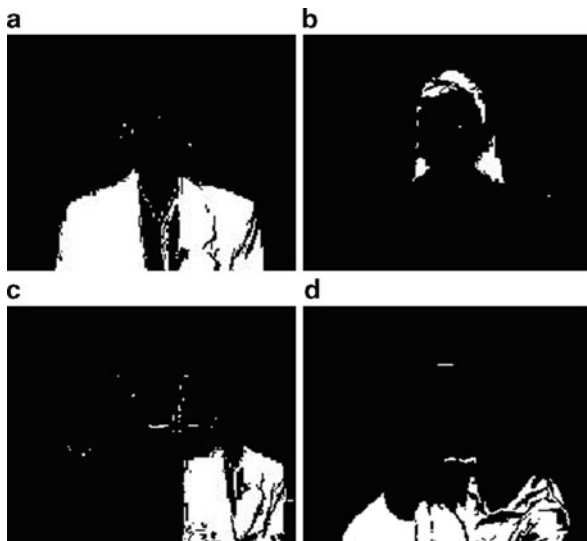
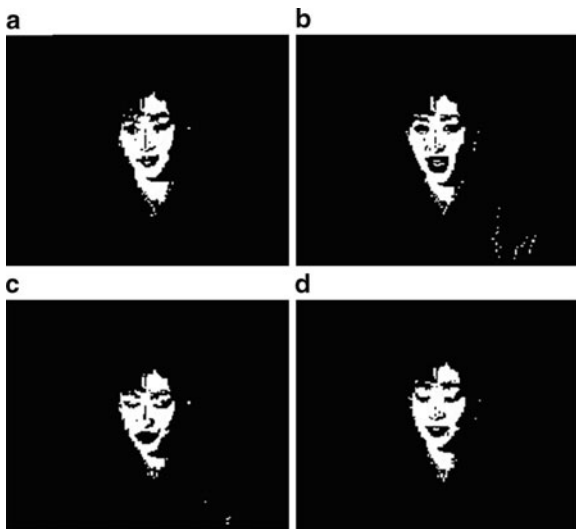


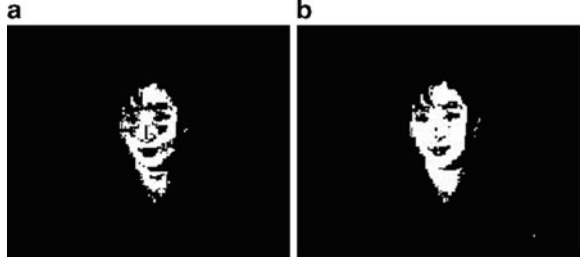
Fig. 2.8 Extracted results of clothes and hair in (a) Akiyo (clothe); (b) Akiyo (hair); (c) News (clothe); (d) Mother-and-Daughter (clothe)

Fig. 2.9 Extracted results with same threshold $k_{1n} = 2$, $k_{2n} = 3$, $k_{1s} = 3$ in different frames: (a) frame 10; (b) frame 30; (c) frame 50; (d) frame 70



order to verify the detection criterion, we need to explore the different deviations, which will significantly influence the formation of signal subspace applying $k_{1n} = 2$, $k_{2n} = 3$, $k_{1s} = 3$ and $k_{1n} = 1$, $k_{2n} = 3$, $k_{1s} = 3$ as described in (2.25), (2.28), and (2.29) to the Akiyo sequence. Figure 2.10b shows that $k_{1n} = 2$ with $\pm 3\sigma_{\lambda_2}$ deviation achieve better segmentation than that with $k_{1n} = 1$ with $\pm 3\sigma_{\lambda_2}$ as in Fig. 2.10a, where the signal deviation is $\pm 3\sigma_{\lambda_1}$ in both cases.

Fig. 2.10 Extracted results by different thresholds with
 (a) $k_{1n} = 1$, $k_{2n} = 3$, $k_{1s} = 3$;
 (b) $k_{1n} = 2$, $k_{2n} = 3$, $k_{1s} = 3$



2.3 Color Object Segmentation Using Adaptive Eigen-Subspaces

In the previous section, we have shown that the eigen-subspace is an effective method for separating the signal and noise components. While segmenting the color video objects, several difficulties could be met. If the object is even originally with the same color, it could show distinct properties under different lightening conditions such as shade. In order to solve such problem, an adaptive eigen-subspace segmentation (AESS) algorithm is proposed [21]. The proposed method can estimate and adaptively adjust the eigenvectors under segmentation procedure. Although the object color is changed with different shade, it is still successfully extracted by using this algorithm. Accompanying with the proposed AESS algorithm, three searching algorithms are used to effectively and efficiently locate the possible pixel. Both AESS and the proposed searching algorithms will be discussed in the following sections.

2.3.1 Adaptive Eigenvector Estimation

The eigen-subspace transformation that we have stated in the previous section was applying the same eigenvectors through the simulation. Sometimes, the eigenvectors need to be adaptively adjusted according to different simulation conditions. It is difficult to segment the color object with different shade by just using the same eigenvectors. We introduce a method that can adaptively adjust the eigenvectors in simulation. As stated in (2.1), we can choose an initial RGB color pixel to form a vector \mathbf{s}_k . Then, the related covariance matrix \mathbf{R}_s can be obtained using (2.2). With covariance matrix, \mathbf{R}_s , we can get the initial eigenvectors \mathbf{w}_0 of the selected color pixel. Also, we can transform the initial RGB color pixels \mathbf{s}_0 to eigen-space using eigenvectors \mathbf{w}_0 as following:

$$\mathbf{y}_0 = \mathbf{w}_0^T \cdot \mathbf{s}_0. \quad (2.32)$$

The term \mathbf{y}_0 is projected vectors of the initial color sample, which forms both the signal space and noise spaces. We can iteratively update the next eigenvectors with \mathbf{y}_0 . The AESS method is illustrated by the following equations as:

$$\mathbf{w}'_k = \mathbf{w}_k \pm 2\mu \mathbf{s}_k \mathbf{y}_k^T. \quad (2.33)$$

where

$$\mathbf{w}_{k+1} = \frac{\mathbf{w}'_k}{\|\mathbf{w}_k\|}. \quad (2.34)$$

and

$$\mathbf{y}_{k+1} = \mathbf{w}_{k+1}^T \mathbf{s}_{k+1}, \quad k = 0, \dots, N. \quad (2.35)$$

The term \mathbf{w}'_k is the eigenvector with deviation $2\mu \mathbf{s}_k \mathbf{y}_k^T$ belongs to previous pixel and \mathbf{w}_{k+1} is the updated eigenvector of the current pixel. The vector \mathbf{s}_k consists of the gray values of red, green, and blue component of the current pixel. The value of the converging parameter μ is small that approximates to 10^{-6} . The plus sign in (2.33) will conduct the equation to reach a maximum value that represents the signal subspace. The minus sign in (2.33) will find the noise subspace that is converged to a minimum value. For iterative computation of the covariance matrix, \mathbf{R}_s , it can be updated as

$$\mathbf{R}_s(k+1) = (1 - \alpha)\mathbf{R}_s(k) + \alpha \mathbf{s}_{k+1} \mathbf{s}_{k+1}^T, \quad (2.36)$$

where $\mathbf{R}_s(k)$ represents the k th covariance matrix of the k th sample color pixel and $\mathbf{R}_s(k+1)$ represents the $(k+1)$ th covariance matrix of the $(k+1)$ th sample color pixel. We can apply following equation to update the eigenvalues without computing the covariance every time.

$$\lambda_{k+1} = (1 - \alpha)\lambda_k + \alpha |\mathbf{y}_{k+1}|^2, \quad (2.37)$$

where λ_k represents the k th eigenvalue and λ_{k+1} represents the $(k+1)$ th eigenvalue. We can find the mean value of the estimated eigenvectors from (2.33). The estimated mean eigenvectors can be represented as follows:

$$E[\mathbf{w}'_k] = E[\mathbf{I} \pm \mu \mathbf{R}_s(k)]E[\mathbf{w}_k], \quad (2.38)$$

where \mathbf{I} is the identity matrix. Inspecting (2.38), we find out that normalized eigenvector \mathbf{w}'_k of previous pixel will approximately equal to eigenvector \mathbf{w}_k of the current pixel. According to this property, we can select μ to adaptively adjust and estimate the eigenvectors of the successive pixels according to (2.33). Using plus and minus sign in (2.38), \mathbf{w}_k will converge to signal plane and noise planes respectively.

2.3.2 Search Algorithms for Desired Color Object

The AESS algorithm is applied to the desired region with first selecting an initial pixel. We take the sampled pixel as the starting point of our searching algorithms. From this starting point, the eigenvectors will be adaptively updated according to different searching routes. Finally, the desired color object will be segmented. The quality of the segmented result is heavily influenced by the shading condition, because the eigenvectors are very sensitive to the color shade. In order to overcome this

shading effect, we proposed three searching algorithms to solve the problems. Using this method, we hope that the segmented result will not be influenced by the background with similar color. With adaptively updating eigen-subspaces according to the search routes, the proposed method can obtain a distinct object. The three search algorithms will be described as follows. The square spiral search (SSS) algorithm is shown in Fig. 2.11. The four quadrant search (FQS) algorithm is shown in Fig. 2.12. The slant horizontal vertical search (SHVS) algorithm is shown in Fig. 2.13. The

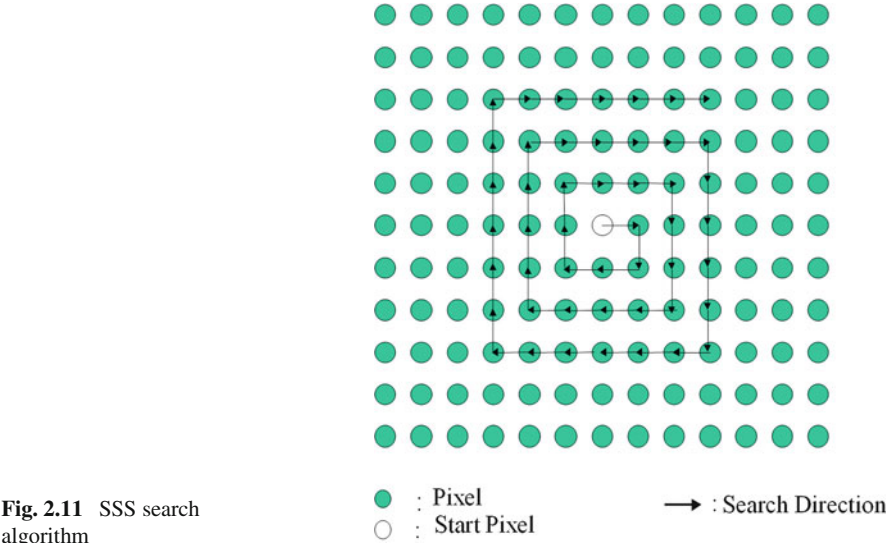


Fig. 2.11 SSS search algorithm

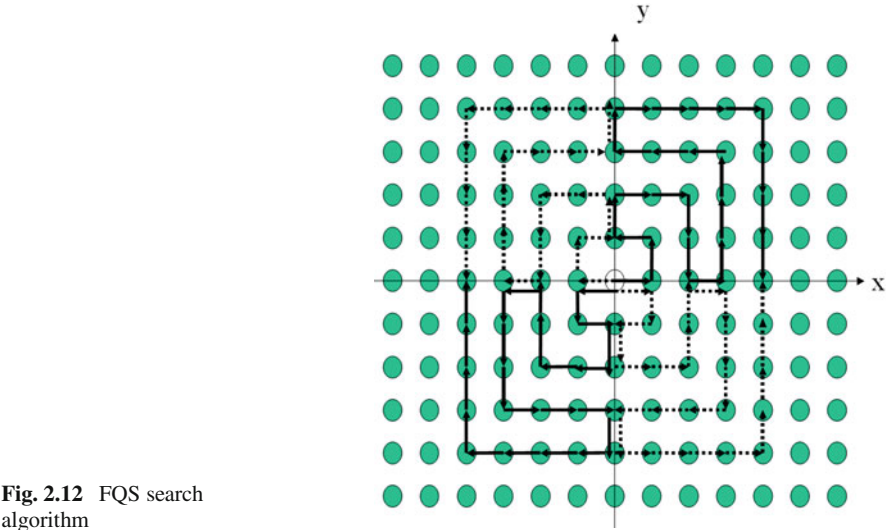
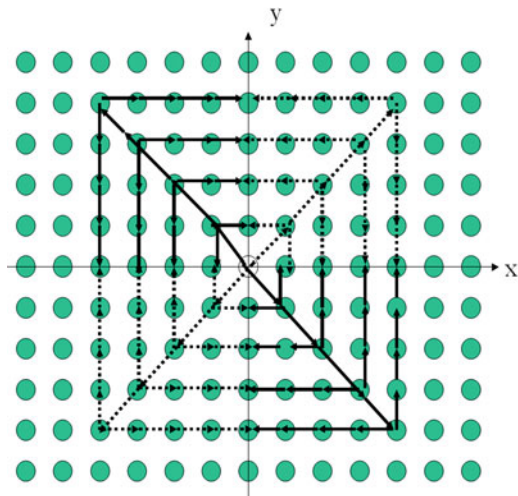


Fig. 2.12 FQS search algorithm

Fig. 2.13 SHVS search algorithm



advantage of the SSS method is suitable to segment the interior part of the object because its color is changing smoothly. But, the SSS method is not good at segmenting the exterior part of the object because the boundaries with different color will be a major problem. The FQS method is searching separately in four quadrants where each area is independent. The advantage of the FQS method is that different color area will not influence each other. The disadvantage of the FQS method is that small area in certain quadrant will be difficult to extract. The main reason is that the eigenvectors are updating too fast and is hard to segment the abrupt color-changing area. The SHVS method takes both advantages from the SSS and the FQS methods that is most suitable to our simulation.

2.3.3 Block Diagram of Adaptive Eigen-Subspace Segmentation Method

Figure 2.14 shows the block diagram of the proposed AEISS method. First, we will sample the chosen pixel and the surrounding eight pixels to obtain the initial eigenvectors as in Fig. 2.15. Then, new eigenspaces, related to the pixel obtained by the searching algorithms as in Figs. 2.11, 2.12, and 2.13 will be formed. Then, we apply the eigen-subspace transformation of the color planes to form the signal and noise planes as shown in Fig. 2.16. Finally, we can use (2.39) to differentiate between the desired and unwanted color pixel. If the pixel is justified as a desired color pixel then the AEISS method will be applied otherwise the algorithm will search next pixel and take the eigenvectors of current pixel as reference. We devise an equation to separate

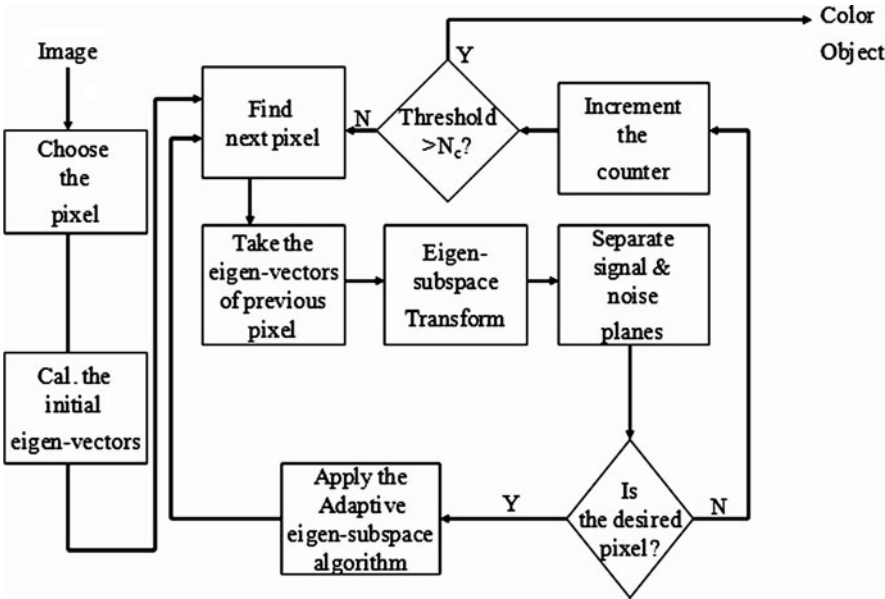


Fig. 2.14 Adaptive eigen-subspace segmentation (AESS) algorithm

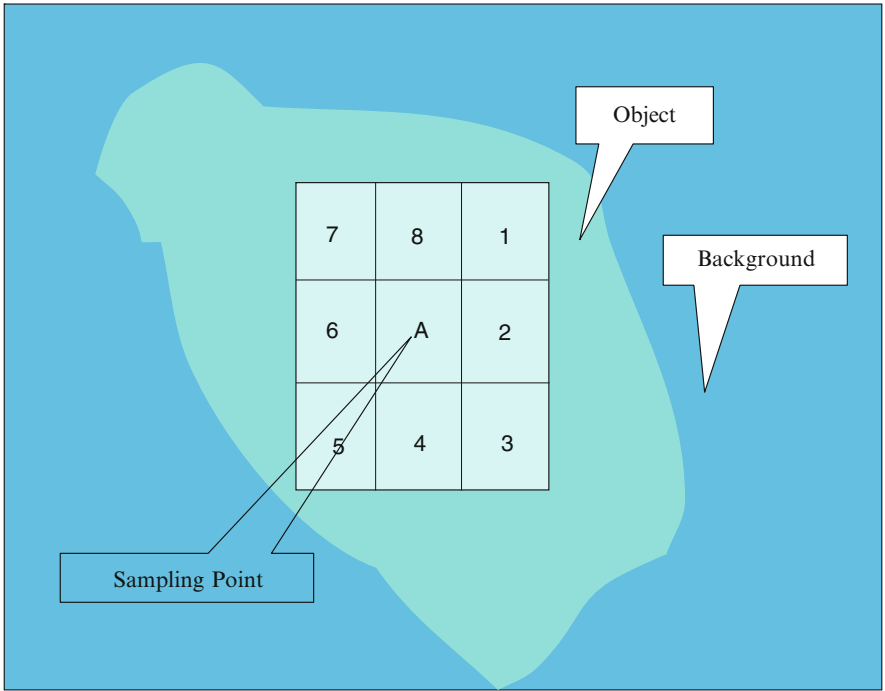


Fig. 2.15 Adaptive eigen-subspace segmentation (AESS) algorithm

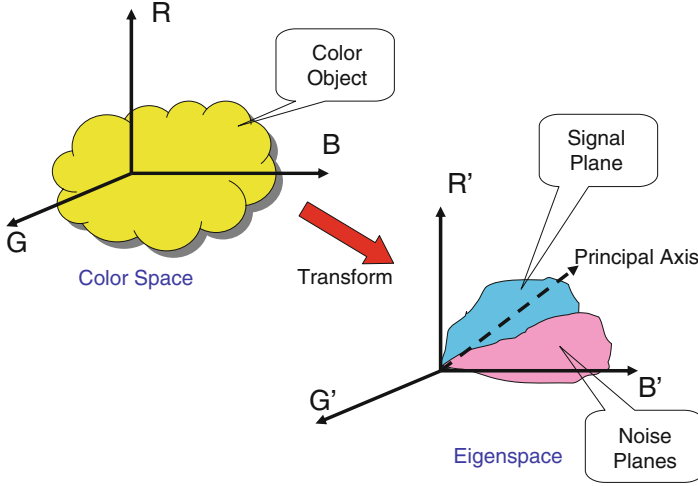


Fig. 2.16 Color space transform to eigenspace

the desired and unwanted pixels that is described as follows:

$$\left| \frac{SA_1^\alpha}{NA_1^\beta \cdot SA_2^\gamma} - \frac{SB_1^\alpha}{NB_1^\beta \cdot NB_1^\gamma} \right| < \Pi, \quad (2.39)$$

where

$$\begin{aligned} SA_1^\alpha &= \mathbf{s}_k \cdot \mathbf{w}_{k-1,s} \\ NA_1^\beta &= \mathbf{s}_k \cdot \mathbf{w}_{k-1,n1} \\ NA_2^\gamma &= \mathbf{s}_k \cdot \mathbf{w}_{k-1,n2} \\ SB_1^\alpha &= \mathbf{s}_{k-1} \cdot \mathbf{w}_{k-2,s} \\ NB_1^\beta &= \mathbf{s}_{k-1} \cdot \mathbf{w}_{k-2,n1} \\ NB_2^\gamma &= \mathbf{s}_{k-1} \cdot \mathbf{w}_{k-2,n2}. \end{aligned}$$

In (2.39), the first term represents the current pixel and the second term represents the previous pixel. The index k represents the current searching location and $k-1$ represents the previous location. The term \mathbf{s}_k and \mathbf{s}_{k-1} represent the current and previous RGB color pixel. The term $\mathbf{w}_{k-1,s}$, $\mathbf{w}_{k-1,n1}$, and $\mathbf{w}_{k-1,n2}$ represent the eigenvectors of the $(k-1)$ th vectors in signal and two noise planes. The terms, $\mathbf{w}_{k-2,s}$, $\mathbf{w}_{k-2,n1}$, and $\mathbf{w}_{k-2,n2}$ represent the eigenvectors of the $(k-2)$ th vectors in signal and two noise planes. The term of SA_1 and SB_1 represent the RGB pixels projected on the signal plane, whereas, NA_1 , NA_2 , NB_1 , and NB_2 are the RGB pixels projected on the noise plane. We find out that is appropriate to take $\alpha = 2$, $\beta = 1$,

and $\gamma = 1$ in simulation. If the difference value of the projection exceeds the threshold Π , then the searching point is set to gray level = 255 otherwise set to 0. Finally, simulation will be terminated if the pixel number of gray = 0 has exceeded certain threshold number N_c .

2.3.4 Simulation Results

Figure 2.17a,b show the result of SSS and FQS algorithm, separately. The simulation sequence Mitq, Clair, Bream, and Students are shown in Fig. 2.18a. The desired color objects that are to be segmented are shown as arrow marks. Applying the SHVS algorithm on the four sequences, the segmented results are shown from Fig. 2.18b. Comparing the Mitq segmentation results in Figs. 2.18 and 2.17, it can be shown that the SHVS algorithm is better than the other two search algorithms. Figure 2.19 shows the segmentation results of Bream sequence with two different parameters according to (2.39): (a) $\alpha = 2, \beta = 1$, and $\gamma = 1$ and (b) $\alpha = 2, \beta = 1$, and $\gamma = 0$.

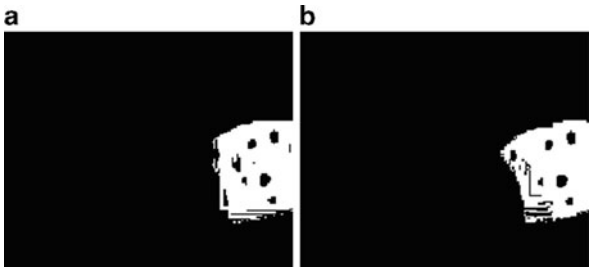


Fig. 2.17 (a) SSS (b) FQS Segmentation Results of Mitq sequence

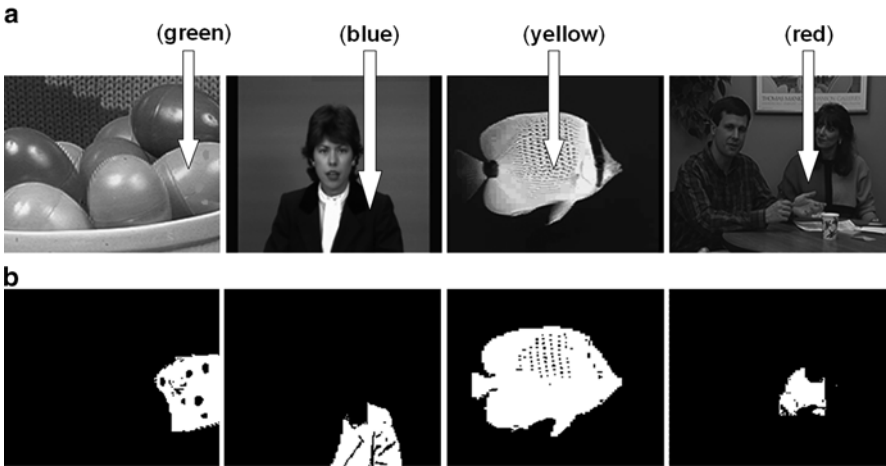
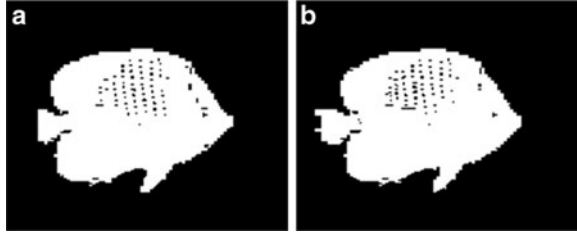


Fig. 2.18 (a) Original sequences (Desired color objects are marked with arrows) (b) Segmentation results by SHVS Methods

Fig. 2.19 Segmentation results with parameters of Bream sequence by AESS method: (a) $\alpha = 2$, $\beta = 1$, $\gamma = 1$; (b) $\alpha = 2$, $\beta = 1$, $\gamma = 0$



2.4 Color Object Segmentation Using Fuzzy C-Means (FCM) with Eigen-Subspace Projection

In this section, two eigen-based FCM methods by combining both PCT and FCM concepts together to achieve effective color segmentation were proposed. In Sect. 2.4.1, conventional FCM algorithms are briefly reviewed. In Sect. 2.4.2, the separated eigen-based FCM (SEFCM) algorithm with the FCM clustering mechanism is separately applied to projections of signal and noise subspaces. Then, a coupled eigen-based FCM (CEFCM) method by introducing an eigen-based membership function embedded in the FCM cluster process is introduced. In Sect. 2.4.3, the simulation results show to verify the proposed methods for any desired color objects segmentation.

2.4.1 Fuzzy C-Means (FCM)

The fuzzy C-means (FCM) algorithm [30–32] is an iterative unsupervised clustering algorithm that robustly adjusts representative centers of each pattern to best partition the data into several distinct classes. The clustering process is accomplished by minimizing an objective function, which is defined by some measure similarity of the data samples. The objective function can be expressed as follows [30, 31]:

$$J_m(\mathbf{U}, \mathbf{V}; \mathbf{X}) = \sum_{q=1}^N \sum_{j=1}^c u_{jq}^m \cdot \text{dist}^2(\mathbf{x}_q, \mathbf{v}_j), \quad (2.40)$$

where N is the number of the data, c is the number of clusters, and scalar m is the arbitrary chosen FCM weighting exponent, which must be greater than one. In (2.40), $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ denotes a set of unlabeled column vectors and $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_c\}$ represents the unknown prototypes, which are known as the cluster centers. The vectors \mathbf{x}_q and \mathbf{v}_j are both k -dimensional real Euclidean space \mathbb{R}^k . Hence, the similarity measurement $\text{dist}(\mathbf{x}_q, \mathbf{v}_j)$ can be specified as either the Euclidean distance or the Mahalanobis distance. The fuzzy C-partition matrix \mathbf{U} is with size of $c \times N$ that its element can be defined as $u_{j,q} \in M_{\text{fcm}}$ as,

$$M_{\text{fcm}} \equiv \left\{ \mathbf{U} = (u_{jq}) | 0 \leq u_{jq} \leq 1, \text{for all } j, q; \right. \\ \left. \sum_{j=1}^c u_{jq} = 1, \text{for all } q; 0 < \sum_{q=1}^N u_{jq} < N, \text{for all } j \right\}.$$

If $\text{dist}(\mathbf{x}_q, \mathbf{v}_j)$ is specified as the Euclidean distance then it can be expressed as

$$\text{dist}(\mathbf{x}_q, \mathbf{v}_j) = \left[\sum_{\alpha=1}^k (x_{q\alpha} - v_{j\alpha})^2 \right]^{\frac{1}{2}}. \quad (2.41)$$

where $\mathbf{x}_{q\alpha}$ and $\mathbf{v}_{j\alpha}$ are the elements in the vector of \mathbf{x}_q and \mathbf{v}_j . If the distance $\text{dist}(\mathbf{x}_q, \mathbf{v}_j)$ is an inner product norm that is called Mahalanobis distance, then, it is expressed as

$$\text{dist}^2(\mathbf{x}_q, \mathbf{v}_j) = \|\mathbf{x}_q - \mathbf{v}_j\|^T \mathbf{A}_j \|\mathbf{x}_q - \mathbf{v}_j\| = \mathbf{Q}_j^T \mathbf{A}_j \mathbf{Q}_j. \quad (2.42)$$

In (2.42), \mathbf{A}_j is a $k \times k$ positive defined matrix derived from the j th cluster. When $\mathbf{A}_j = \mathbf{I}$, (2.42) is equal to the Euclidean norm as specified in (2.41). For $m > 1$ and $\mathbf{x}_q \neq \mathbf{v}_j$, the objective function $J_m(\mathbf{U}, \mathbf{V}; \mathbf{X})$ may lead to a minimum if the following equations hold:

$$u_{jq} = \frac{(dist_{jq})^{\frac{-2}{m-1}}}{\sum_{i=1}^c (dist_{iq})^{\frac{-2}{m-1}}} \quad \forall j, q. \quad (2.43)$$

and

$$\mathbf{v}_i = \frac{\sum_{q=1}^N (u_{jq})^m \mathbf{x}_q}{\sum_{q=1}^N (u_{jq})^m} \quad \forall i. \quad (2.44)$$

The similarity measure terms dist_{jq} and dist_{iq} specified in (2.43) can be defined as either (2.41) or (2.42) with respective cluster center \mathbf{v}_i or \mathbf{v}_j . Unlike traditional classification algorithms, the FCM algorithm assigns all object patterns to each cluster in fuzzy fashions. Each pattern associated with a belonging specified by membership grades between 0 and 1. The fuzzy membership value describes how close or accurate a sample resembles an ideal element of a population. The imprecision caused by vagueness or ambiguity is characterized by the membership value. Inclusive of the concept of fuzziness, the FCM algorithm computes each class center more precisely and with higher robustness to the noise. The procedures of the FCM algorithm [30, 31] are enlisted as follows:

1. Initialization: Fix the number of cluster c and feature coefficient m , set iteration loop index $t = 0$, and select initial cluster centers.

We are randomly select c initial cluster centers from the space as $\mathbf{v}_j^{(0)}$, for $j = 1, 2, \dots, c$. Initialized $\mathbf{U}^{(0)}$.

2. Sampling: Choose total N data samples \mathbf{x}_q for $q = 1, 2, \dots, N$ from the image. It is performed by clicking the mouse on the image.

3. Calculating the fuzzy cluster centers: Compute all cluster centers $\{\mathbf{v}^{(t)}\}$ using $\mathbf{U}^{(t-1)}$ with the equation specified in (2.9).
4. Update membership function $\mathbf{U}^{(t)}$: Update $\mathbf{U}^{(t)}$ using $\mathbf{v}^{(t)}$ with the equation specified in (2.8).
5. Check convergence condition: Check the previous defined convergence behavior as Δ by computing

$$\Delta = \left| \mathbf{v}^{(t)} - \mathbf{v}^{(t-1)} \right|. \quad (2.45)$$

6. If $\Delta < \varepsilon$ or a preset loop count N_t is reached then terminate; otherwise set $t = t + 1$ and go to Step 3, where ε is the preset terminating criterion.

In (2.45), the superscript t denotes the number of iterations. If the changes of the class centers are less than a predefined criterion ε that means the objective function $J_m(\mathbf{U}, \mathbf{V}; \mathbf{X})$ is no longer decreasing. The final segmentation result is achieved.

To improve orientation sensitivity (OS), Schmid in [33] suggested a modified FCM algorithm (OSFCM) by modifying \mathbf{A}_j described in (2.42) as:

$$\mathbf{A}_j = \mathbf{V}_j^T \mathbf{L}_j \mathbf{V}_j, \quad (2.46)$$

where \mathbf{L}_j denotes the diagonal matrix containing the inverse of eigenvalues and \mathbf{V}_j represents the unitary matrix lining up the corresponding eigenvectors of the fuzzy covariance matrix \mathbf{C}_j^x for the j th cluster. The fuzzy covariance matrix for the j th cluster \mathbf{C}_j^x is given by

$$\mathbf{C}_j^x = \frac{1}{N} \sum_{q=1}^N u_{jq}^m (\mathbf{x}_q \mathbf{x}_q^T - \mathbf{v}_j \mathbf{v}_j^T). \quad (2.47)$$

From simple matrix derivations, it is obvious that $\mathbf{A}_j = (\mathbf{C}_j^x)^{-1}$.

2.4.2 Eigen-Based FCM Algorithms

In this section, we combine the FCM classification with eigen-subspaces projection together to achieve effective color segmentation. By using eigenvectors, we can transform the original color space into the modal coordinate system of the desired color as

$$\mathbf{z}_q = [\mathbf{w}_q \ \mathbf{w}_2 \ \mathbf{w}_3]^T \mathbf{x}_q = [\phi_q \ \varphi_q \ \psi_q]. \quad (2.48)$$

Now, the first-principal elements, ϕ_q for $q = 1, 2, \dots, N$ specify the signal subspace whereas the second and the third elements φ_q and ψ_q and build the noise subspaces. The vector \mathbf{x}_q is defined as in (2.40). With signal and noise subspaces, we develop two eigen-based FCM detection procedures, the separate eigen-based FCM (SEFCM) and the coupled eigen-based FCM (CEFCM) methods. The main procedures of eigen-based FCM are shown in Fig. 2.20. First, we compute the covariance matrix \mathbf{R}_s of the desired color samples from the RGB color planes followed by

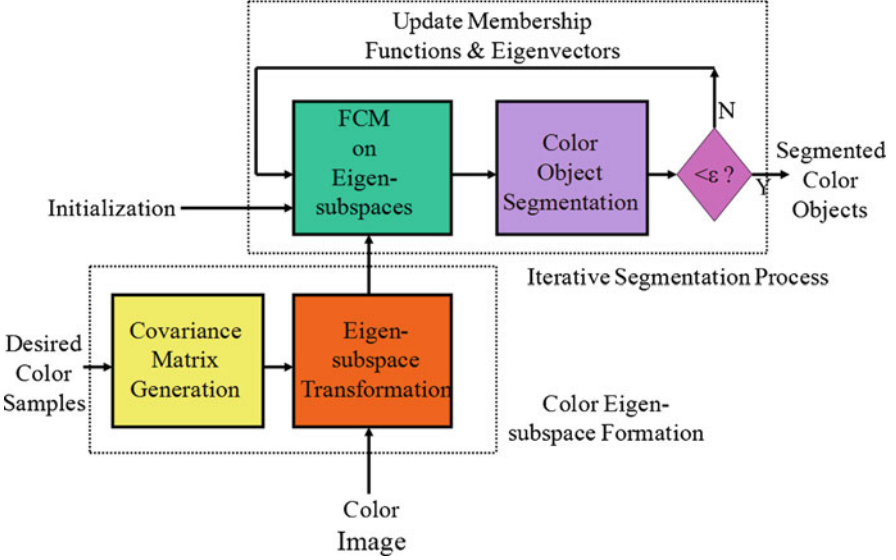


Fig. 2.20 The proposed eigen-based FCM algorithms

the eigen-analysis procedures to obtain the eigenvectors. Then, we perform the eigen-subspace transformation of image color planes with (2.48). Finally, the iterative segmentation process with updated membership functions will be applied on the eigen-subspaces. Although the eigenvectors are generated by the selected color samples, we still need to adjust the eigen-subspaces to achieve more satisfactory segmentation results. We are iterative in adjusting the eigen-subspaces with new eigenvectors obtained from new covariance matrix. The detail algorithms of SEFCM and CEFCM will be addressed in Sects. 2.4.2.1 and 2.4.2.2.

2.4.2.1 Separate Eigen-Based FCM (SEFCM) Method

In this section, we have proposed the SEFCM algorithm to separately consider the signal and noise planes. During simulation, we iteratively construct new covariance matrices that are similar to (2.47) by using the eigen-subspace data \mathbf{z}_q instead of \mathbf{x}_q in color image. The expression of the covariance matrix can be stated as follows:

$$\mathbf{C}_j^z = \frac{1}{N} \sum_{q=1}^N u_{jq}^m \mathbf{z}_q \mathbf{z}_q^T. \quad (2.49)$$

The color objects will be extracted after the objective function reaches a minimum. With the help of the eigenvalues, we can obtain the represented segmented color objects with respective to the signal and noise subspaces. Following, with a simple logical “AND” operation on both results, we can obtain the segmentation of

the desired video objects correctly. In the SEFCM, we modify the matrix \mathbf{L}_j as in (2.46) that is suitable to extract the signal and noise subspaces. For extracting the signal space, we can rewrite \mathbf{L}_j as follows:

$$\Gamma_j = \begin{pmatrix} \left(\frac{\lambda_{2,j} + \lambda_{3,j}}{2}\right)^{-1} & 0 & 0 \\ 0 & \lambda_{1,j}^{-1} & 0 \\ 0 & 0 & \lambda_{1,j}^{-1} \end{pmatrix}. \quad (2.50)$$

Similarly, we can extract the noise planes by using the following matrix:

$$\Gamma_j = \begin{pmatrix} \lambda_{1,j}^{-1} & 0 & 0 \\ 0 & \left(\frac{\lambda_{2,j} + \lambda_{3,j}}{2}\right)^{-1} & 0 \\ 0 & 0 & \left(\frac{\lambda_{2,j} + \lambda_{3,j}}{2}\right)^{-1} \end{pmatrix}. \quad (2.51)$$

We adopt (2.50) to extract the signal plane by using $\lambda_{1,j}^{-1}$ to suppress the noise terms. In (2.51), we use $\lambda_{1,j}^{-1}$ to suppress the signal terms in order to obtain two noise planes. We can modify the membership function of (2.43) as follows:

$$u_{jq} = \frac{[(\mathbf{z}_q - \mathbf{v}_j)^T \mathbf{A}_j (\mathbf{z}_q - \mathbf{v}_j)]^{\frac{-2}{m-1}}}{\sum_{\beta=1}^c [(\mathbf{z}_q - \mathbf{v}_\beta)^T \mathbf{A}_\beta (\mathbf{z}_q - \mathbf{v}_\beta)]^{\frac{-2}{m-1}}}, \quad (2.52)$$

where $\mathbf{A}_j = \mathbf{V}_j^T \Gamma_j \mathbf{V}_j$ and $\mathbf{A}_\beta = \mathbf{V}_\beta^T \Gamma_\beta \mathbf{V}_\beta$ related to class j and β , respectively. For class β , the index j appeared in (2.50) and (2.51) should changed to β . The detailed procedures of the SEFCM are shown in Fig. 2.19 and illustrates as follows:

1. Sample few desired color object blocks.
2. Compute the covariance matrix and obtain the eigenvectors according to (2.2).
3. Transform the color images to signal and noise subspaces with eigenvectors as (2.48).
4. Initialize the modified membership value and center of each cluster. With iterative updating of the covariance matrices using (2.49), apply FCM to extract the segmentation results related to signal and noise planes separately. Either segmenting on signal or noise planes, we apply (2.50) or (2.51) to the new membership function (2.52) during the FCM classification procedures.
5. Perform logical operation on the results obtained from Step 4.

2.4.2.2 Coupled Eigen-Based FCM (CEFCM) Method

In order to efficiently segment the desired color objects, we devise a coupled eigen-based FCM (CEFCM) algorithm (Fig. 2.21). In considering signal and noise planes

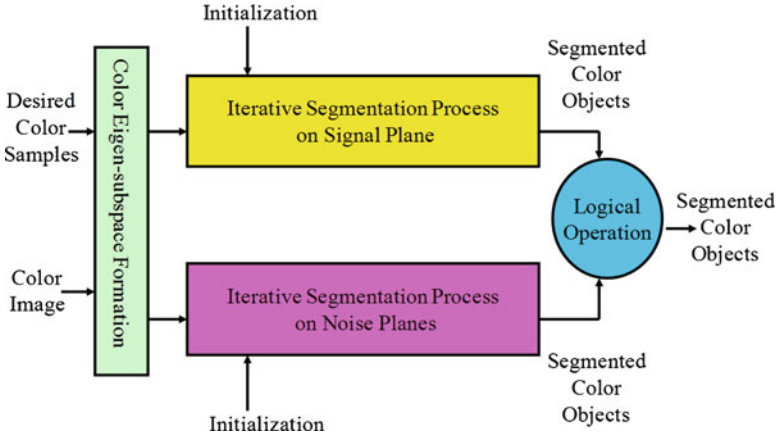


Fig. 2.21 Signal flow diagram of proposed SEFCM algorithm

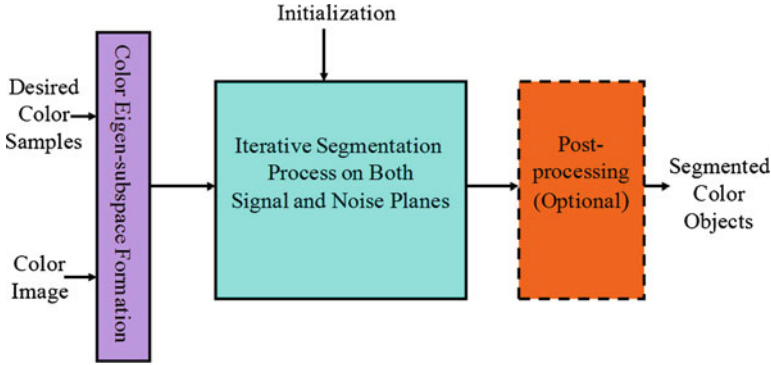


Fig. 2.22 Signal flow diagram of proposed CEFCM algorithm

together, the CEFCM adopts three dimensional eigen-subspaces data for classification. The function block diagram of the CEFCM is shown in Fig. 2.22. Similar to SEFCM, we also construct new covariance matrices that are similar to (2.47) by using the eigen-subspace data \mathbf{z}_q in stead of \mathbf{x}_q in color images. In view of statistical inference and fuzzy property, we can construct a new covariance matrix for the j th cluster center as

$$\hat{\mathbf{C}}_j^z = \frac{1}{\sum_{q=1}^N u_{jq}^m} \sum_{q=1}^N u_{jq}^m \mathbf{z}_q \mathbf{z}_q^T. \quad (2.53)$$

It is not necessary to iteratively rebuild the covariance matrix and construct the new eigen-subspaces from the original color images because the color objects are selected under our inspection. We can gradually adjust the direction of principal axes by using the already built eigen-subspaces so that large amount of transformation computations can be saved. In updating procedures, we adopt the new covariance

matrix as described in (2.53), where the belongings are treated as the weightings of $\mathbf{z}_q \mathbf{z}_q^T$ that is formed by the data in the eigen-subspaces. From covariance matrix $\hat{\mathbf{C}}_j^z$, we can obtain class j 's eigenvector $\mathbf{w}_{j,i}$ and the corresponding eigenvalue $\lambda_{j,i}$, where index i denotes the i th component because we order the eigenvalues as

$$\lambda_{j,1} \geq \lambda_{j,2} \geq \lambda_{j,3}. \quad (2.54)$$

Then, the j th cluster center of signal term in the CEFCM can be expressed by the principal component

$$\mathbf{v}_{j,1} = \sqrt{\lambda_{j,1}} \mathbf{w}_{j,1}. \quad (2.55)$$

The second and the third components are treated as the noise terms. The similarity measure related to the j th cluster center expressed as Euclidean distance between \mathbf{z}_q and $\mathbf{v}_{j,1}$ now becomes the orthogonal projection to the noise eigenvectors. The smaller of $\|\mathbf{z}_q - \mathbf{v}_{j,1}\|$ means that \mathbf{z}_q and $\mathbf{v}_{j,1}$ are closer to each other. Based on eigen properties, the smaller projection to both $\mathbf{w}_{j,2}$ and $\mathbf{w}_{j,3}$ with respect to $\sqrt{\lambda_{j,2}}$ and $\sqrt{\lambda_{j,3}}$ indicates that \mathbf{z}_q and $\mathbf{v}_{j,1}$ are closer. The measure of $\|\mathbf{z}_q - \mathbf{v}_{j,1}\|$ related to $\mathbf{w}_{j,2}$ and $\mathbf{w}_{j,3}$ is equivalent to the projection of \mathbf{z}_q to the normalized noise eigenvectors, which can be expressed as $\frac{1}{\sqrt{\lambda_{j,2}}} \mathbf{w}_{j,2}$ and $\frac{1}{\sqrt{\lambda_{j,3}}} \mathbf{w}_{j,3}$, so that the membership of the q th sample can be modified as follows:

$$u_{jq} = \frac{\left(\frac{1}{\lambda_{j,2}} \|\mathbf{z}_q^T \mathbf{w}_{j,2}\|^2 + \frac{1}{\lambda_{j,3}} \|\mathbf{z}_q^T \mathbf{w}_{j,3}\|^2 + \frac{1}{\lambda_{j,1}} (\|\mathbf{z}_q^T \mathbf{w}_{j,1}\|^2 - \lambda_{j,1}) \right)^{\frac{-2}{m-1}}}{\sum_{\beta=1}^c \left(\frac{1}{\lambda_{\beta,2}} \|\mathbf{z}_q^T \mathbf{w}_{\beta,2}\|^2 + \frac{1}{\lambda_{\beta,3}} \|\mathbf{z}_q^T \mathbf{w}_{\beta,3}\|^2 + \frac{1}{\lambda_{\beta,1}} (\|\mathbf{z}_q^T \mathbf{w}_{\beta,1}\|^2 - \lambda_{\beta,1}) \right)^{\frac{-2}{m-1}}}. \quad (2.56)$$

The success of extracting video objects depends on the proportion of three eigenvalues. Inspecting (2.56), we have adjusted the iterating processes near the cluster center in the signal subspace according to its eigenvalue to eliminate too bright or too dark circumstances. In our experiments, we take fuzzy weighting $m = 3$, the class number equals to 6 and feature number equals to 3. Observing the simulation results, we can obtain more satisfactory ones by just considering the principal plane and the strongest noise plane. In this case, we set $\lambda_{j,1} = \lambda_{j,3} = 1$, $\lambda_{\beta,1} = \lambda_{\beta,3} = 1$, and $\lambda_{j,2} \rightarrow \infty$; $\lambda_{\beta,2} \rightarrow \infty$ with $c = 2$ in (2.56).

After obtaining the segmentation results, almost all the desired color pixels can be found. The few remaining noise pixels can be easily removed by any post-processing procedure. The detailed procedures for the CEFCM are listed as follows:

1. As Step 1 in Sect. 2.4.2.1.
2. As Step 2 in Sect. 2.4.2.1.
3. As Step 3 in Sect. 2.4.2.1.
4. Initialize the membership function and cluster centers. Later, we jointly consider three eigen-subspaces and iteratively update the covariance matrix with (2.53).

With newly found eigenvectors and eigenvalues, we apply (2.55) and (2.56) to perform the FCM classification processes.

5. (Optional). Post-processing procedure is optional for smoothing the results from Step 4. In our experiments, we do not intent to use any post-processing procedures in order to show the inherent classified capability of this algorithm.

2.4.3 Simulation Results

We directly apply the conventional FCM to these four sequences in the RGB color planes. The simulation results are shown in Fig. 2.23. Without any threshold determination, the desired objects obtained by the traditional FCM are better than those obtained by the PCT method. However, the segmented results still contain many unwanted noises.

Without any other assistance, Fig. 2.24 shows the simulation results by applying conventional FCM to the transformed planes, which are performed by the KL projections. Compared to Fig. 2.23, some improvements are achieved. The KL projections obtained from the desired color samples can translate the image data to the desired working space in more compaction form. It is reasonable to apply the segmentation efforts on the eigen-subspaces. Theoretically, results of Figs. 2.6 and 2.24 should be identical because of the linear transformation between color-space and eigenspace. The difference of the results shown in these two figures may be due to initial data distribution and class centers.

For comparison, we also apply OSFCM algorithm [33] to the eigen-subspaces. Figure 2.25 shows the segmented images obtained from the OSFCM method. Although all the main objects can be detected, the objects with near color are also

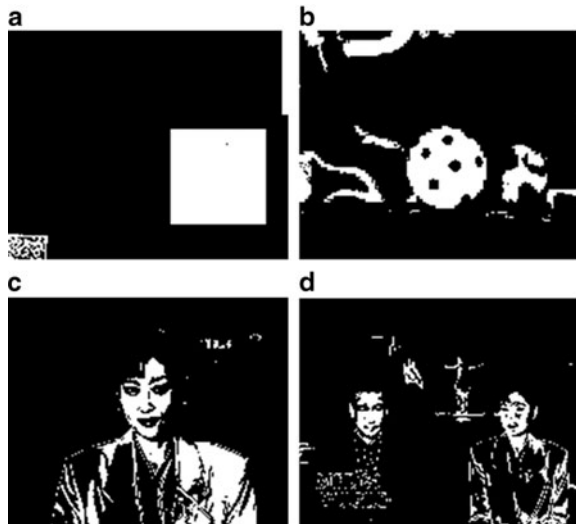


Fig. 2.23 Segmented images obtained by the conventional FCM directly applying on R,G,B planes: (a) Mosaic; (b) Ball; (c) Akiyo; (d) News sequences

Fig. 2.24 Segmented images obtained by the conventional FCM applying on the KL transformed color spaces: (a) Mosaic; (b) Ball; (c) Akiyo; (d) News sequences

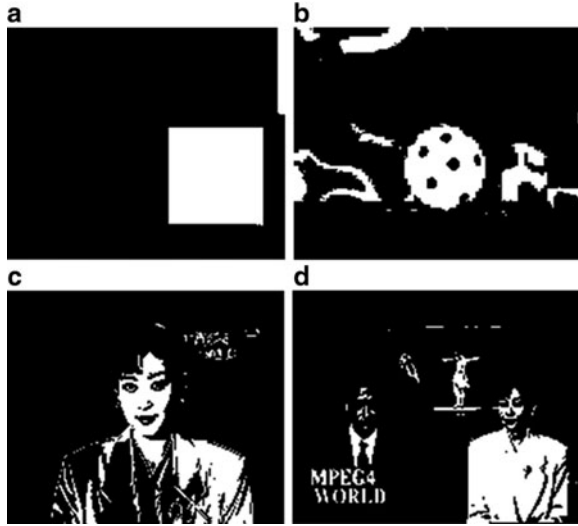
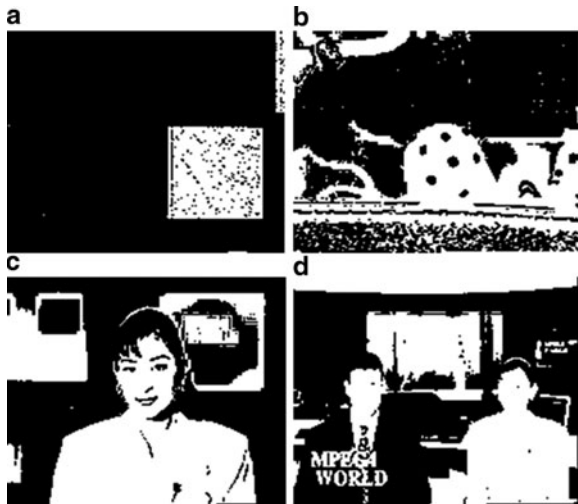


Fig. 2.25 Segmented images obtained by the OSFCM method: (a) Mosaic; (b) Ball; (c) Akiyo; (d) News sequences



extracted. Similar to the PCT approach, the OSFCM method does not separately use the noise-subspace. The color objects with high signal-subspace projects will be erroneously included in the similar color pixels.

Figure 2.26 shows the results obtained by the SEFCM. We can find out that most of the noise has been removed compared to the previous obtained results. The major defeat of the SEFCM method appears on the clothes of Akiyo since both signal and noise subspace projections do not perform simultaneously. In order to improve the performance, the CEFCM algorithm adopts the signal and noise subspace

Fig. 2.26 Segmented images obtained by the SEFCM method: (a) Mosaic; (b) Ball (c) Akiyo; (d) News sequences

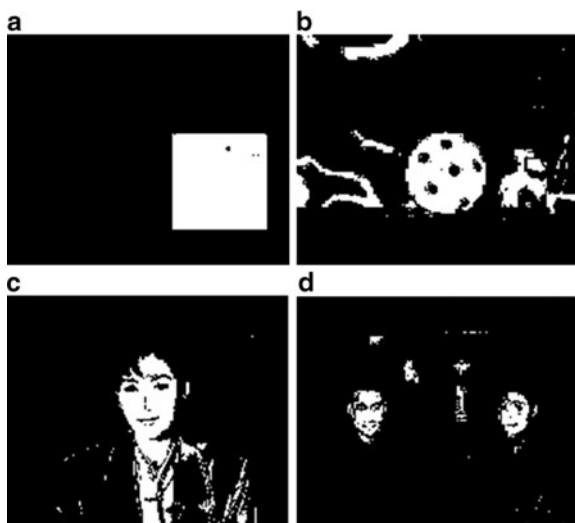
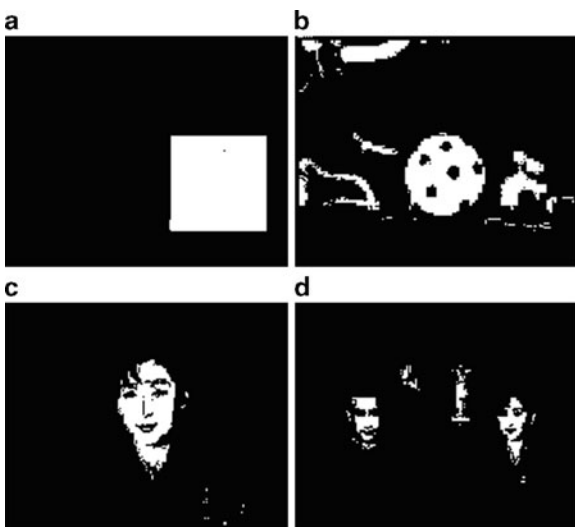


Fig. 2.27 Segmented images obtained by the CEFCM method: (a) Mosaic; (b) Ball; (c) Akiyo; (d) News sequences



projections together. Figure 2.27 shows the segmented images obtained from the CEFCM method. It is obvious that the CEFCM method outperforms the other color object segmentation algorithms.

2.5 Conclusions

Video object segmentation has been recognized as a main technology to achieve the content-based coding proposed in the MPEG-4 standard. In this chapter, we proposed color eigenspace segmentation methods to extract the desired objects.

After theoretical analyses, the color eigen-structure segmentation algorithm uses both signal and noise subspaces effectively. Without any pre-processing process, we can precisely detect the desired color and preserve small significant features. If the segmented results still contain some undesirable pixels, of course, we can easily remove them once we further introduce temporal or motion information. The proposed color segmentation algorithm performs successfully for single color objects. When the desired object is with more than two colors, for example color texture or color pattern, we should apply our algorithm several times accordingly. The final video object planes will be the union of all the segmented results.

In Sect. 2.3, adaptive eigen-subspace segmentation (AESS) algorithm to locally extract the desired objects has proposed. Three search methods are used in the AESS algorithm which is SSS, FQS, and SHVS. Using this algorithm, we can extract the desired color objects even if the background has the same color as the object. The simulation results show that the AESS algorithm can achieve better segmentation results. It has a localized segmentation capability.

In Sect. 2.4, we use the conventional FCM combined with the eigenspace projections concept to develop two segmentation algorithms. The first algorithm uses the eigen-structure combined with the FCM method and the second algorithm simply applies threshold on the eigen-spaces according to the statistical properties. The method with FCM is very effective to segment desired objects without considering any threshold. The drawback of the first method is that it costs large amount of computation time. The second method adopting statistical analysis is faster than the first one but needs to select different threshold values according to different sequences.

Considering signal and noise projections, the SEFCM method shows its effectiveness comparing to use the PCT or the FCM algorithm alone. In order to achieve satisfactory simulation results, we further suggest the CEFCM algorithm to improve the segmentation performance. Compared to the conventional FCM method and the OSFCM method, we found that the SEFCM and CEFCM achieve the best segmentation performance, which is robust and less susceptible to the noise. Further integrated with spatial and temporal information, our proposed algorithms can achieve even better results in the future works. The difference between AESS algorithm and the other method is that our proposed method can be used to segment the desired object at will. Especially, when several identical color objects exist at the same frame, our algorithm can extract only the designated object out. Using color eigen-subspace to segment the object has been proven to be an effective method but it maybe failed when the color shade is present. In addition to the normal color distribution conditions, our algorithm is most suitable to segment the object with color shade.

References

1. ISO/IEC JTC1/SC29/WG11, N2201 (1998) Text for ISO/IEC FCD 14496-1 Systems
2. ISO/IEC JTC1/SC29/WG11, N2502 (1998) Information technology - coding of audio-visual objects, Part 2: visual

3. Lee MC, Chen W, Lin CB, Gu C, Markoc T, Zabinsky SI, Szeliski R (1997) A Layer Video Object Coding System Using Sprite and Affine Motion Model, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 130–145.
4. Zheng H, Blostein SD (1995) Motion-based object segmentation and estimation using the MDL principle, *IEEE Transactions on Image Processing*, vol. 4, no. 9, pp. 1223–1235.
5. Chang MM, Tekalp AM, Sezan MI (1997) Simultaneous motion estimation and Segmentation, *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1326–1333.
6. Chu CC, Aggarwal JK (1993) The integration of image segmentation maps using region and edge information, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 12, pp. 1241–1252.
7. Wani MA, Batchelor BG (1994) Edge-region-based segmentation of range images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 314–319.
8. Kaup A, Aach T (1998) Coding of segmented images using shape-independent basis functions, *IEEE Transactions on Image Processing*, vol. 7, no. 7, pp. 937–947.
9. Williams PS, Alder MD (1996) Generic texture analysis applied to newspaper segmentation, *IEEE International Conference on Neural Networks*, vol. 3, no.3–6, pp. 1664–1669.
10. Waked B, Bergler S, Suen CY, Khoury S (1998) Skew detection, page segmentation, and script classification of printed document images, *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5, no. 11–14, pp. 4470–4475.
11. Littmann E, Ritter H (1997) Adaptive color segmentation-a comparison of neural and statistical methods, *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 175–185.
12. Liu J, Yang YH (1994) Multiresolution color image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 7, pp. 689–700.
13. Shafarenko L, Petrou H, Kittler J (1998) Histogram-based segmentation in a perceptually uniform color space, *IEEE Transactions on Image Processing*, Vol.7, No. 9, pp. 1354–1358.
14. Gonzalez RC, Woods RE (1992), *Digital Image Processing*, MA: Addison-Wesley
15. Murase H, Nayar SK (1994) Illumination panning for object recognition using parametric eigenspaces, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 12, pp. 1219–1227.
16. Xiuping J, Richards JA (1999) Segmented principal components transformation for efficient hyperspectral remote-sensing image display and classification, *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no.1, Part-2, pp. 538–542.
17. Kerfoot IB, Bresler Y (1997) Theoretical analysis of multispectral image segmentation criteria, *IEEE Transactions on Image Processing*, vol. 8, no. 6, pp. 798–820.
18. Kumaresan R, Tufts DW (1983) Estimating the angles of arrival of multiple plane waves, *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 1, pp. 134–139.
19. Hu B, Gosine RG (1997) A new eigenstructure method for sinusoidal signal retrieval in white noise: estimation and pattern recognition, *IEEE Transactions on Signal Processing*, vol. 45, no. 12, pp. 3073–3083.
20. Kaveh M, Barabell AJ (1986) The statistical performance of the MUSIC and the minimum-norm algorithms resolving plane waves in noise, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, no. 2, pp. 331–341.
21. Xu XX, Hao SS, Yang JF (2000) Video object color segmentation using adaptive eigen-subspace, *Proceedings of 2000 workshop on internet distributed systems*, May 11–12, National Cheng Kung University, Tainan, Taiwan, pp. 122–127.
22. Yang JF, Hao SS, Chung PC (2002) Color image segmentation using fuzzy C-means with eigen-subspace projections, Vol. 82, *Signal Processing*, pp. 461–472.
23. Wan X, Kuo CCJ (1998) A New Approach to Image Retrieval with Hierarchical Color Clustering, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 628–643.
24. Antoszczyszyn PM, Hannah JM, Grant PM (1998) Reliable tracking of facial features in semantic-based video coding, *IEEE Proceedings-Vision, Image and Signal Processing*, vol. 145, no. 4, pp. 257–263.
25. Special Issue on Very Low Bit Rate Video Coding (1994) *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 3.

26. Cai J, Goshtasby A, Yu C (1998) Detecting human faces in color images, Proceedings of the International Workshop on Multi-Media Database Management Systems, pp. 124–131, OH, USA.
27. Chai D, Ngan KN (1998) Locating facial region of a head-and-shoulders color image, Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition, pp. 124–129, WA, Australia.
28. Hartung J, Jacquin A, Pawlyk JS, Rosenberg J, Okada H, Crouch PE (1998) Object-Oriented H.263 Compatible Video Coding Platform for Conferencing Applications, IEEE Journal on Selected Areas in Communications, vol. 16, no. 1, pp. 42–55.
29. Lee JH, Chang BH, Kim DS (1994) Comparison of colour transformations for image segmentation, Electronics Letters, vol. 30, no. 20, pp. 1660–1661.
30. Bezdek JC (1981) Pattern recognition with fuzzy objective function algorithm, Plenum, New York.
31. Tsoukalas LH, Uhrig RE (1997) Fuzzy and Neural Approaches in Engineering, John Wiley & Sons, Inc.
32. Haykins S (1999) Neural Networks-A Comprehensive Foundation, 2nd edition, Prentice Hall International, Inc.
33. Schmid P (1999) Segmentation of digitized dermatoscopic images by two-dimensional color clustering, IEEE Transactions on Medical Imaging, Vol. 18, No. 2, pp. 164–171.

Video Segmentation and Its Applications

Ngan, K.N.; Li, H. (Eds.)

2011, XIV, 163 p., Hardcover

ISBN: 978-1-4419-9481-3