,   o

Moving Object Segmentation with Accurate Boundary Using Color and Motion

Kuk-Jin Yoon and In-So Kweon[O]

Department of Electrical Engineering, KAIST

kjyoon@covra1.kaist.ac.kr, iskweon@kaist.ac.kr.

Abstract

The edge and motion are the main features that human visual system (HVS) perceives intensively. Therefore, it is very important to obtain accurate boundary of moving object coinciding with the boundary that HVS perceives in object segmentation process. This paper proposes an algorithm for the segmentation of the moving object with accurate boundary using color and motion focusing on HVS perception in the general image sequence. The proposed algorithm is composed of three parts: color segmentation, motion analysis, and region refinement and merging part. In the color segmentation phase, K-Means algorithm is used in consideration of the sensitivity of the human color perception to get the accurate boundaries coinciding with the boundaries that HVS perceives. The global and local motion estimation are performed in parallel with color analysis. As the result of color and motion analysis, boundary and motion information of each region are obtained. After that, Bayesian clustering using color and motion provides more accurate boundary for each region although the color contrast between objects and background is low. In the final stage, regions are merged taking into account their motion. The experimental results of the proposed algorithm show the accurate moving object boundary coinciding with the boundary that HVS perceives.

1. Introduction

Object-based video analysis describes the contents of the video focusing on the information about the objects, such as shape, color, intensity and gesture, and etc. Object-based technologies take an important role in video analysis. Moving object segmentation is the heart of the object-based technologies and can be applied to various applications such as object recognition, object tracking, object-based video indexing and object-based video coding. So, it has been studied for the last few decades in computer vision and image coding area. As the result, many algorithms and systems have been proposed and developed. In spite of these efforts, however, it is the one of the most difficult problem yet.

The best known mechanism for moving object segmentation is the Human Visual System (HVS). Moving object segmentation is the high level analysis. The performance evaluation is answered by the person. So, in the moving object segmentation process, it is important to extract the object boundary coinciding with HVS because HVS has high sensitivity about motion and edge.

In this paper, we present a motion segmentation algorithm that adaptively combine color and motion based on the sensitivity of the human visual system.

2. Previous Work

It is assumed that the object is composed of a few regions. Also, the region is composed of the pixels that have coherent feature. So, image segmentation is the fist step to extract the objects in image.

Image segmentation algorithm can be classified into three groups according to the used feature: spatial segmentation, temporal segmentation and spatio-temporal segmentation. Spatial segmentation uses the spatial feature only, such as color, intensity, and texture. The result of spatial segmentation shows accurate region boundary without any contextual information. Therefore, it is difficult to extract the contextually meaningful object. Temporal segmentation can segment the contextually meaningful object because it uses temporal information. But, the motion, which is the temporal feature, is difficult to be estimated accurately; the result of temporal segmentation shows inaccurate region boundaries. To define and segment the contextually meaningful moving object in the image sequence, it is necessary to use the spatial feature and temporal feature together. Spatio-temporal segmentation uses the two kinds of features to

segment moving objects.

In many previous researches, the spatial features, such as color, intensity and texture, determine object boundary [1, 2, 3]. Input images are segmented into regions by spatial feature. Then motion of each region is estimated to merge the region having similar motion. So, the boundary is determined by the spatial feature only and the location of the moving region is estimated by temporal feature only. In simple images, color boundary is accurate. But, if the contrast of the spatial feature of the object and background is low, some part of object and background may be merged. To overcome this limitation, approaches using spatial and temporal feature simultaneously to determine the boundary and location of moving object have been proposed [4, 5]. These approaches use the joint similarity measure of spatial and temporal feature between the pixel and region to assign the region label to each pixel. The joint similarity measure is assumed the form of the weighted linear combination of spatial similarity and temporal similarity. So, the determination of optimal weight value is difficult and the boundary location may be changed depending upon the weight.

3. Proposed Algorithm

We propose an algorithm that preserves the accurate boundary of moving object in consideration of HVS. Figure (1) shows the overall structure of the proposed algorithm composition. In this paper, we assume that the object is composed with a few color regions having coherent motion. So, we use color as the spatial feature and motion as the temporal feature.
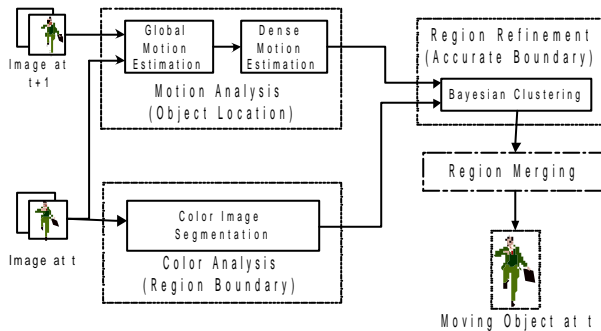

Figure 1 Proposed Algorithm

The Proposed algorithm consists of three components: the color segmentation and motion estimation and region refinement process. Because color and motion are the main cues that HVS has high sensitivity, the preservation of accurate color boundary and accurate motion estimation is important. After the color image segmentation and motion analysis, region boundaries are refined by using the color and motion information of each region together in the post-processing part. So, region boundaries are refined to coincide the boundaries that the HVS perceives. In the final phase, regions that have similar motion vectors are merged.

3. 1 Color Image Segmentation

There are many approaches to color image segmentation. Among the many color segmentation algorithms, we use the K-Means algorithm, which is known as a powerful method to deal with the large color pixel set [7, 8, 9].

In color image segmentation, it is important to preserve the human perceivable color boundary. But, conventional clustering methods don't have reflected the characteristics of the human visual system. First, the Euclidean distance between two color points in a color space cannot accurately reflect the difference of the HVS color perception. According to the position of color points in the color space, the same Euclidean distance does not mean the same color difference. Next, they do not consider the sensitivity of the human color perception. The HVS shows different color perception sensitivity according to the color distribution in the image domain [6]. In conventional approaches, the cluster centers, shapes in the color space and the boundaries of color regions in the image domain are determined by the color values in the image only. As the result, the region boundary doesn't coincide the human perceivable color boundaries and some boundaries are missed.

In addition to these problems, the K-Means algorithm has the fatal limitation: how to determine the number of clusters. That is very important for the accuracy of the segmentation and the efficiency of clustering processing.

To solve these problems, we propose a color segmentation algorithm that specifically takes into account the characteristics of the human color perception. We use the CIELab color space, in which Euclidean distance of two colors is proportional to the difference that human visual system perceived. We also introduce the color weight to consider the human color perception. Human visual system is more sensitive to the color in homogeneous region than complex region as mentioned earlier. Figure (2) shows the flower garden image with two test color points: one is in the sky and the other is in the flower garden. Although the colors of test points are similar with colors of neighboring points, the HVS perceives the color of the point in the sky more vividly.

Figure 2 Flower Garden Image with Test Color Points

To take into account this phenomenon, we assign a weight value to each pixel color. The pixels surrounded by a region with a homogeneous color have larger weights and the pixels in the complex colored region have smaller weights. As the result, the cluster center and shapes are formed to preserve the more sensitive color boundaries

that are analogous with HSV. Color weight means the color sensitivity of each pixel color considering the neighboring color distribution.

To assign a color weight to each pixel, we use a local mask centered at each pixel. We first calculate the average color and the color variance in the local mask. The average color is given by

$$[L_{avg} \quad a_{avg} \quad b_{avg}] = \frac{1}{N}[\sum_{j \in M} L_j \quad \sum_{j \in M} a_j \quad \sum_{j \in M} b_j] \quad (1)$$

where N is the number of pixels in the local mask M. In CIELab color space, a small Euclidean distance between two color points is proportional to the difference that HVS perceives. But, a large Euclidean distance has no meaning but only large difference in HVS. Therefore, color difference in CIELab color space can be modeled as;

$$D_{i,j} = 1 - e^{-E_{i,j}/g} \quad (2)$$

where g is the normalized factor and $E_{i,j}$ is the Euclidean distance in CIELab color space,

$$E_{i,j} = \sqrt{(L_i - L_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2} \quad (3)$$

The color variance in the local mask is given by

$$v_i = \frac{1}{N}\sum_{j \in M} D_{avg,j}^2$$
$$= \frac{1}{N}\sum_{j \in M}(1 - e^{-\sqrt{(L_{avg}-L_j)^2 + (a_{avg}-a_j)^2 + (b_{avg}-b_j)^2}/g})^2 \quad (4)$$

The color variance in the local mask is the measure how the color pattern varies in the local area. A large color variance means that there is frequent color pattern variation in the neighboring area. So, large weight is assigned to the pixel that has a small color variance. We use a Gaussian kernel to generate the weight

$$W_i = e^{\frac{-v_i^2}{2s^2}} \quad (5)$$

To determine the number of clusters, we use the calculated weight values. The number of clusters must be determined taking into account the distribution of color components in the color space and image domain. If the average value of the color weights is large, it means that there are large parts of homogeneous region in image domain. In this case, a small number of clusters is enough. For the large variance of the weight value, a large number of clusters is needed. We use the following equation to determine the number of clusters.

$$K(m_w, v_w) = M \times [e^{-m_w * a} + (1 - e^{-v_w * b})] \quad (6)$$

where M denotes the maximum number of clusters to be used and a, b are some constants value to adjust the optimal number of clusters. We set a, b as 4,2 respectively. In the K-Means process, color weight is used to determine the center of each cluster. The objective function and the center of each cluster are given by

$$J = \sum_{i=1}^{K}\sum_{x \in S_i}\|x - m_i\|^2 w(x) \quad (7)$$

$$m_i^{t+1} = \frac{\sum_{x \in S_i} x \times w(x)}{\sum_{x \in S_i} w(x)} \quad (8)$$

3.2 Motion Estimation

Motion is also an important cue to HVS. In the image, there are two kinds of motion: global motion and local motion. Global motion is due to the motion of the camera, such as panning, tilting, zoom-in and zoom-out, and local motion is due to moving object in the scene. To extract the moving object, global motion is compensated first. After global motion compensation, the motion in the image is wholly due to object motion.

To estimate and compensate the global motion, we use the 6-parameter affine model as the motion model and the NDIM (Normalized Dynamic Image Model) as the image model [10].

The NDIM is devised to overcome the intensity change due to illumination change or object motion. The NDIM models the intensity change 1st order linear equation. The relationship between corresponding pixels can be simplified using ICA (Intensity Conservation Assumption);

$$\frac{(I_1(\mathbf{x}) - m_1(\mathbf{x}))}{s_1(\mathbf{x})} = \frac{(I_2(\mathbf{x} + \Delta\mathbf{x}) - m_2(\mathbf{x} + \Delta\mathbf{x}))}{s_2(\mathbf{x} + \Delta\mathbf{x})} \quad (9)$$

where m and s denote the average and standard deviation of intensity in a small window.

After global motion compensation, there are local motion components only in the image. This is due to the object motion completely. We use the robust method proposed by Black and Anandan [11] to compute dense optical flow.

3.3 Region Boundary Refinement

The initial boundary of each region is determined by color. If the color contrast between object and background is high, color is a good cue to determine the boundary of moving region. In this case, color boundary coincide the boundary that HVS perceives. But, if the contrast between object and background is low, the resulting boundary may be inaccurate and some parts of object may be merged with the background. Region boudnary refinement is done for the accurate object boundary and acuurate motion of each region.

We propose the statistical approach, Bayesian clustering, to refine the region boundaies. It is assumed that the pixels in a same region have coherent feature. Therefore, pixels in a same region shows similar statistical property when the regions are modeled as the statistical models. Let x_{ij} denote a statistically independent 5×1 feature vector consisting of three color components and two motion components. Also let the pixel set be $S = \{x_{ij}, i = 1,2,...,W, j = 1,2,...,H\}$. The number of regions M is already computed by the color image segmentation process. Then, we can partition the set S into

the M regions, $R = \{R_1, R_2, ..., R_M\}$. The pixels in the same region k are described by the probability density $p_k(x_{ij} \mid \mathbf{q}_k)$, where p_k is known function and $\mathbf{q}_k$ is a parameter vector whose values have to be determined. We models the color and motion as the Gaussian random variables, so p_k is given as Gaussian.

If the label of the pixel in the border of the region is incorrect, the probability density value will be low because that pixel has different statistical property from the region. The region label of that pixel must be changed if the pixel has larger probability density value when it is assumed that that pixel belongs to the neighboring region. In other words, the pixels must have a region label in which has the largest probability density value. Therefore, if we assume priors of $\mathbf{q}$ and R be uniform, then the MAP estimates (R*, $\mathbf{q}$*) are given by

$$(\text{R*}, \mathbf{q}\text{*}) = \arg\max_{R,\mathbf{q}} P(S \mid R, \mathbf{q}) \qquad (10)$$

Since the pixels are independent, the joint density of S has the following form

$$P(S \mid R, \mathbf{q}) = \prod_{k=1}^{M} (\prod_{x_{ij} \in R_k} p_k(x_{ij} \mid \mathbf{q}_k)) \qquad (11)$$

By applying natural logarithm, we can get following relation.

$$\ln(P(S \mid r, \mathbf{q})) = \ln(\prod_{k=1}^{M} (\prod_{x_{ij} \in R_k} p_k(x_{ij} \mid \mathbf{q}_k)))$$
$$= \sum_{k=1}^{M} \sum_{x_{ij} \in R_k} \ln(p_k(x_{ij} \mid \mathbf{q}_k)) \qquad (12)$$

So, we can get another form of eq.(10) like

$$(\text{R*}, \mathbf{q}\text{*}) = \arg\max_{R,\mathbf{q}} \sum_{k=1}^{M} \sum_{x_{ij} \in R_k} \ln(p_k(x_{ij} \mid \mathbf{q}_k)) \qquad (13)$$

For a fixed $\mathbf{q}$, the R which maximizing the eq. (13) can be obtained using

$$R_k = \{x_{ij}, \ln p_k(x_{ij} \mid \mathbf{q}_k) > \ln p_t(x_{mn} \mid \mathbf{q}_t),$$
$$\forall k \neq t, t = 1, 2, ..., M\} \qquad (14)$$

where k=1, 2, .., M. Similarly for a fixed R, the maximizing value of $\mathbf{q}$ is unique and it can be obtained using

$$\mathbf{q}_k = \max \sum_{x_{ij} \in R_k} \ln(p_k(x_{ij} \mid \mathbf{q}_k) \qquad (15)$$

where k=1, 2, ..,.M.

where N_k denotes the number of pixels in the region.

Because we have initial region and region labels of the pixels already, we test the statistics of pixels in the border of the regions only. Starting from an initial R_k and $\mathbf{q}_k$ computed in the previous stage, Bayesian clustering steps are like as follows.

$$\mathbf{m}_k = \frac{1}{N_k} \sum_{x_{mn} \in R_k} x_{mn}, \qquad \mathbf{s}_k^2 = \frac{1}{N_k} \sum_{x_{mn} \in R_k} (x_{mn} - \mathbf{m}_k)^2 \qquad (16)$$

The initial R_k and $\mathbf{q}_k$ are obtained by the result of the color segmentation result and motion estimation result in previous stage. So, the Bayesian clustering steps are like as follows.

Region Refinement

Step 1. Initialize R_k and $\mathbf{q}_k$, k=1, 2, ..,.M

Step 2. Given $R_k^{(i)}$, Compute $\mathbf{q}_k^{(i)}$ using eq.(16)

Step 3. Given $\mathbf{q}_k^{(i)}$, Compute $R_k^{(i+1)}$ using eq.(14)

Step 4. If $R_k^{(i)} = R_k^{(i+1)}$, stop the iteration. Else go to step 2.

As the result of region refinement, we can get the accurate region boundaries coinciding with the boundaries that HVS. Also, accurate motion information is obtained by refining the boundary of each region. We assumed that the object is composed with a few color regions having coherent motion.

To extract the complete object, regions that have a similar motion are considered to belong to one object. Because we are interested in moving regions only, we must detect the regions that have independent motion first. We determine the moving region by applying a threshold to the magnitude of motion vector. After detecting the moving regions, we merge the regions having a similar motion vector in to a larger one.

4. Experimental Results

Figure (3) shows the experimental result of color image segmentation for the flower garden. For the conventional algorithm, we assume that all pixel colors have same weights. From the result, we can observe that the proposed methods provides more accurate boundary in uniform color regions such as the sky.

(a) Flower garden
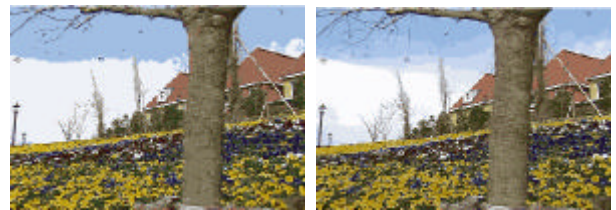
(b) Conventional algorithm   (c) Proposed algorithm

Figure 3 Color Image Segmentation Results

Figure (4) shows the extracted motion for two consequent images, which include sudden intensity changes. As mentioned earlier, there are two kinds of

motion. The global motion is due to the panning of the camera and local motion is due to the swing motion of the player. To estimate the global motion, input images are transformed to the NDIM images first. In Figure (4-b), only the moving object parts are detected as outliers after global motion compensation.


(a) Input images
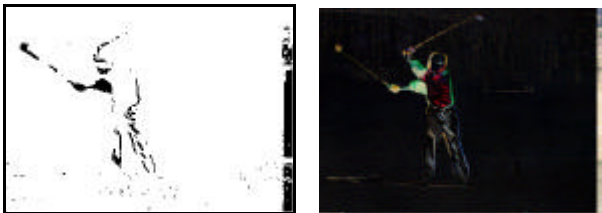
(b) Detected outlier   (c) Difference image after GMC

Figure 4 Global Motion Estimation Result

Another results for the Foreman sequence are shown in Figure (6). In these input images, the color of helmet is very similar with the background color as shown in Figure (6). As you can see, some parts of the object and the background are merged where the color contrast between the object and the background are vary low. Some parts of the hat of the object in the foreground are merged with the wall in the background because they have the similar color. That is an unavoidable result in color segmentation. In parallel with color segmentation, motion in the image is estimated. Figure (6-c) shows the estimated optical flow and relative disparity. The low intensity denotes the large disparity of the pixel.

(a) Input images
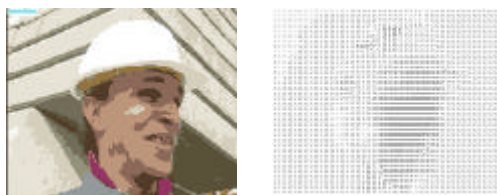
(b) Color segmentation   (c) Optical flow

Figure 5 Color Segmentation and Motion Estimation Results

Figure (6) shows the region refinement result. We use

the color and motion information together to refine the border of the regions. So, we can overcome the limitation of the color segmentation. As the result, we can get the accurate boundaries coinciding the boundaries that HVS perceives because the boundaries are refined by two main visual cues. Also, the accurate motion of each region is obtained.

(a) Before Refinement   (b) After refinement

Figure 6 Region Refinement Result

Figure (7) shows the final result by the proposed algorithm and a conventional algorithm, which uses the spatial feature for boundary information and temporal feature for moving region location respectively. The conventional algorithm shows erroneous boundaries and some parts of foreground region are lost. But, the proposed algorithm shows reasonably accurate boundary that HVS perceives by using two visual cues; motion and color.
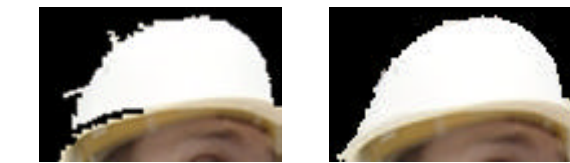

(a) Conventional algorithm   (b) Proposed algorithm

Figure 7 Moving Object Segmentation Result

Figure (8) shows the area that the contrast of the foreground and background is low minutely. Also, some experimental results are also shown in Figure (9).
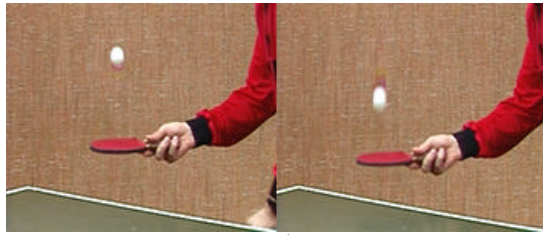

(a) Low contrast boundary

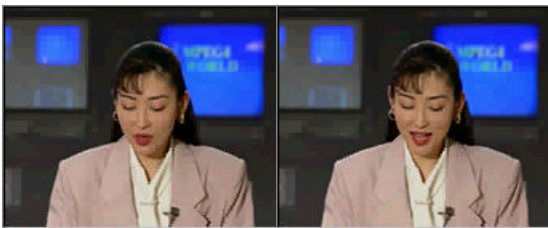(b) Conventional algorithm   (c) Proposed algorithm

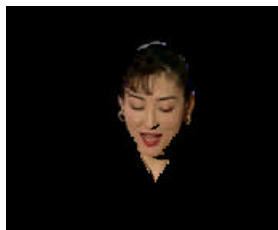**Figure 8 Segmentation Result at Low Contrast Area**

(a) Input images

(b) Segmentation result

(c) Input images

(d) Segmentation result

Figure 9 Moving Object Segmentation Results

5. Conclusion and Further Works

In this work, we proposed a moving object segmentation algorithm with accurate boundary, that human visual system perceivable, focusing on the two main visual cues, color and motion. Human visual system has different sensitivity to color and is also very sensitive the boundary information. We introduced the color weight in clustering processing to assign more weight to the color that HVS shows high sensitivity. The experimental result of the color segmentation shows the more accurate region boundaries in uniform colored regions that HVS perceives more vividly. But, color is insufficient to extract the accurate boundary because the color contrast may be low between foreground and background at the border of the object. Therefore, another visual cue, motion, is used together with color to extract the proper boundary for HVS. The final result of the proposed algorithm gives an accurate segmentation result with HVS perceivable boundaries.

But, proposed algorithm has some limitation. In the final phase, we detect the moving region by thresholding the magnitude of motion vector. It is difficult to set the optimal threshold value. Also, region merging is performed by using only motion information. So, some parts of the object may be missed because some parts of the object may be stationary.

References

[1] V. Rehrmann, "Object Oriented Motion Estimation in Color Image Sequence", European Conference on Computer Vision, Vol. 1, Page(s) : 704-719, 1998.

[2] M. Hoetter and R. Thoma, "Image Segmentation Based on Object Oriented Mapping Parameter Segmentation", Signal Processing, Vol. 15, No. 3, pp.315-334, October 1988.

[3] J. Guo, J.W. Kim and C-C. J. Kuo, "Fast Video Object Segmentation Using Affine Motion And Gradient-Based Color Clustering", IEEE Second Workshop on Multimedia Signal Processing, Page(s): 486 –491, 1998.

[4] K. W. Song, E. Y. Chung, J. W. Park, G. S. Kim, E. J. Lee and Y. H. Ha, " Video Segmentation Algorithm using a Combined Similarity Measure for Content-based Coding", Proceedings of the Picture Coding Symposium, Page(s) :261-264, 1999.

[5] J. G. Choi, S.W. Lee and S. D. Kim , "Spatio-temporal video segmentation using a joint similarity measure, IEEE transactions on circuits and systems for video technology, Vol. 7, No. 2. April 1997.

[6] B. A. Wandell, "Color Appearance: the Effects of Illumination and Spatial Patterns", Proc. Nat. Acad. Sci., USA, v 90, p. 1494-1501, 1993.

[7] T. Uchiyang and Michael A. Arbib, "Color Image Segmentation Using Competitive Learning", IEEE Transactions on Pattern Analysis and Machine Intelligent, vol. 16, No. 12, December 1994.

[8] N. Kehtarnavaz, J. Monaco, J. Nimtschek and A. Weeks, "Color Image Segmentation Using Multiscale Clustering", Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation, 142-147, 1998.

[9] L. Lucchese, S.K. Mitra, "An Algorithm for Unsupervised Color Image Segmentation", IEEE Second Workshop on Multimedia Signal Processing, Page(s): 33 –38, 1998.

[10] Y. S. Moon and I. S. Kweon, "Robust Dominant Motion Estimation Algorithms against Local Linear Illumination Variations", Technical Report of Robotics and Computer Vision Lab, Dept. of EE, KAIST, Jun. 1999.

[11] M. J. Black and P. Anandan, "The Robust Estimation of Multiple Motions: Parametric and Piecewise-smooth Flow Field", Computer Vision and Image Understanding, Vol. 63, No. 1, January, pp. 75-104, 1996

[12] S. Sista and R. L. Kashyap, "Bayesian Estimation for Multiscale Image Segmentation", Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 6, Pages: 3493-3496, 1999.