Video Segmentation by Event Detection: A Novel One-Class Classification Approach

Mahesh Venkata Krishna, Paul Bodesheim, and Joachim Denzler

Computer Vision Group,
Friedrich Schiller University Jena,
07743 Jena, Germany
{mahesh.vk,paul.bodesheim,joachim.denzler}@uni-jena.de
http://www.inf-cv.uni-jena.de

Abstract. Segmenting videos into meaningful image sequences of some particular activities is an interesting problem in computer vision. In this paper, a novel algorithm is presented to achieve this semantic video segmentation. The goal is to make the system work unsupervised and generic in terms of application scenarios. The segmentation task is accomplished through event detection in a frame-by-frame processing setup. For event detection, we use a one-class classification approach based on Gaussian processes, which has been proved to be successful in object classification. The algorithm is tested on videos from a publicly available change detection database and the results clearly show the suitability of our approach for the task of video segmentation.

1 Introduction

A major goal of computer vision is to make computers do at least some of the vision-based tasks that are currently tedious or difficult for humans to perform. In many real-world applications it is often the case that there is a large corpus/stream of videos with interesting events sparsely spread over it. It then becomes an exhausting task to extract interesting events from the huge amount of data. The aim of this work is to achieve this interesting event extraction by segmenting the input video into various semantic *phases*. Applications of this vary widely, e.g. in surveillance, most of the time nothing abnormal happens but suddenly there is an important activity like breach of rules. In microscopic videos recording microbial activities, interesting events happen sparsely over time and it is a huge waste of manpower to keep it under constant observation. In such situations, automated event extraction is a very important tool.

In addition, video segmentation gets us into important theoretical issues such as what defines an interesting event and how can a machine *guess* by itself what an interesting event may be. As definitions of events are application dependent, a generic definition of an event can only be stated as : “Something that is not normal in the video”, i.e. something *novel*. This leads us to the main contribution of this work: the application of one-class classification (OCC) algorithms to generic unsupervised video segmentation. In our approach, detecting temporal novelties in a video is a big step towards video segmentation. To perform one-class classification, we look among the various approaches that have been proposed in the field of object classification. There exist various approaches based on one-class classification techniques, such as Gaussian process regression (GPR) [3], support vector data description (SVDD) [9], one-class SVM [8], or Parzen density estimation [6]. One can clearly see that the OCC setup matches our problem scenario: a model of normality (or known patterns) has to be built by clustering in the feature space and novelty is declared in case an outlier is met in the testing phase. We show in the following sections how one-class classification techniques can be used for video segmentation.

The remainder of this paper is organized as follows. First, in section 2, we review related work in the field of video segmentation. A brief discussion of their relative advantages and shortcomings is provided. In section 3, we review one of the most prominent OCC techniques, namely Gaussian process regression. We then present our video segmentation approach in section 4 together with the explanation how to use OCC techniques for event detection in videos. The results on the thermal videos of the CVPR change detection dataset [2] are presented in section 5 highlighting the suitability of our approach. A summary of our findings and suggestions for future research directions conclude the paper.

2 Previous Work

The work by Koprinksa and Carrato [4] provides a good survey of video segmentation techniques based on a very diverse range of theoretical concepts and for various applications. Most of the algorithms presented there concentrate on directly finding the differences between frames through some distance measure between feature vectors of consecutive frames. A threshold is then applied to this distance to achieve a segmentation. This approach yields reasonable results but is based on directly finding inter-frame differences without *modeling* the underlying scenario. Thus, they can not detect semantic changes, e.g. approaches based on global color histograms may not detect events such as a person bending or falling as it is not likely that there is a significant change in the color histogram.

Liu et al. [5] have presented an approach based on the perceived motion energy feature, where optical flow vectors in each frame are averaged and multiplied with a factor arising out of the dominant direction of motion. These features are then clustered to form segments of the video corpus. This method is very useful for videos containing a lot of motion, but when events happen that do not alter the motion profile of the frames (e.g. color changes), it is likely to fail.

Therefore, we need a generic framework for video segmentation without assumptions of a specific application. The next section provides the theoretical aspects of our proposed approach based on OCC.

3 One-Class Classification Techniques

In this section, we briefly describe the idea behind OCC as well as one of the most prominent techniques, i.e. the Gaussian process regression framework presented by [3].

3.1 The Task of One-Class Classification

In an OCC scenario, there are only training samples $\boldsymbol{X} = \left(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right)$ of a single class available. Thus, all of them have the same constant label, e.g. $\boldsymbol{y} = \mathbf{1} = (1, 1, \ldots, 1)^{\mathsf{T}}$. This class is often referred to as target class and the corresponding samples as target data or target set [9]. The aim is to find an appropriate description of the class distribution to distinguish this single class from every other possible and currently unknown class. Therefore, a novelty score should be inferred for each test sample $\boldsymbol{x}^*$ such that a large score indicates strong membership to the target class. If this score is below a certain threshold, the test sample will be treated as an outlier not belonging to the estimated distribution. The following methods allow for a suitable modeling with samples that only stem from a single class.

3.2 Gaussian Process Regression

The Gaussian process framework is a well-known probabilistic methodology that is successfully used for tasks such as regression and classification [7]. In the case of Gaussian process regression (GPR), outputs $y(\boldsymbol{x})$ are assumed to be generated according to a latent function f and a noise term ε:

$$y(\boldsymbol{x}) = f(\boldsymbol{x}) + \varepsilon \quad . \tag{1}$$

Following a Bayesian framework, output values of unknown samples $\boldsymbol{x}^*$ are predicted probabilistically by marginalizing over both latent function values and noise. While this is in most cases infeasible to realize exactly, a few assumptions make the prediction tractable:

1. Latent functions f are drawn from a Gaussian process prior with zero mean, and covariance function κ.
2. The noise term is assumed to be normally distributed: $\varepsilon \sim \mathcal{N}(0, \sigma_{\mathsf{n}}^2)$.

Using these assumptions, the predictive distribution over output values is normally distributed, i.e. $y^* | \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}^* \sim \mathcal{N}(\mu_*, \sigma_*^2)$, where moments μ_* and σ_*^2 can be computed in closed form. The work of [3] shows how GPR can be used to solve OCC problems. The authors propose using either the predictive mean μ_* (GPR-Mean) or negative variance $-\sigma_*^2$ (GPR-Var) as novelty scores:

$$\mu_* = \boldsymbol{k}_*^{\mathsf{T}} \left(\boldsymbol{K} + \sigma_{\mathsf{n}}^2 \boldsymbol{I}\right)^{-1} \mathbf{1} \quad \text{and} \tag{2}$$

$$-\sigma_*^2 = -\left(\boldsymbol{k}_{**} - \boldsymbol{k}_*^{\mathsf{T}} \left(\boldsymbol{K} + \sigma_{\mathsf{n}}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_* + \sigma_{\mathsf{n}}^2\right) \quad , \tag{3}$$

where $\boldsymbol{K} = \kappa\left(\boldsymbol{X}, \boldsymbol{X}\right), \boldsymbol{k}_* = \kappa\left(\boldsymbol{X}, \boldsymbol{x}^*\right), \boldsymbol{k}_{**} = \kappa\left(\boldsymbol{x}^*, \boldsymbol{x}^*\right)$, and $\boldsymbol{I}$ the unit matrix.

We have decided to choose the GPR approach, because it is shown by [3] that Gaussian processes are superior to the SVDD approach of [9] in visual object categorization. Since one-class SVM leads to equivalent results compared to SVDD [8], the GPR approach is our method of choice. Having this OCC technique on hand, we explain how to use it for video segmentation in the next section.

4 Video Segmentation by Event Detection

In this section, we first focus on our idea of using OCC methods for video segmentation. Afterwards, we present the algorithm of our OCC approach. The last part of this section is about the features we use to obtain proper video segmentation.

4.1 The Idea of Our Approach

The goal of this work is to semantically segment videos into *meaningful* image sequences. To achieve such a partition, an approach similar to work flow segmentation can be used, where different *phases* of the video are marked based on their semantic content. In each time step, we model the *current* situation with a OCC model and then look for special events (novelties) in the consecutive frames.

Thus, in our approach we want to detect events that lead to a change of the current phase. This is carried out by using OCC methods as explained in the following section. In the end, we are able to evaluate the resulting segmentation of the whole sequence which comes from the detected events.

4.2 Our One-Class Classification Approach

Let us assume that there are features available for each frame stored in a specific feature vector. The feature extraction methods will be explained in section 4.3. We start with learning a one-class model (Sec. 3) using the features of the first F frames of the video. Here, we assume that there is no event within these first frames and assign them with phase count 1. In most real-life situations, a few frames in the order of 30 to 60 correspond to 2-3 seconds, where it is reasonable to assume that no interesting event happens. For every consecutive frame, we evaluate the learned model to obtain its novelty score. This is done until the score of a frame drops below a specified threshold T. If this is the case, we have detected an event leading to a phase change.

Assuming that such events, which indicate a phase change, are sparsely spread over time (i.e. do not happen closely, with a gap of at least F frames between them), we learn a new one-class model with the features of the next F frames. The unlabeled previous frames are assigned with the phase count of the old model and we update the phase count of the current model. The video sequence is segmented this way, completely unsupervised. An overview of our approach can be seen in Figure 1.

Note that our approach does not need a training step using manually labeled sequences to learn a suitable model. Moreover, it can be directly applied to any video sequence since the model is learned on-the-fly within the sequence that should be segmented. We may only have to adjust the parameters F and T as well as method specific parameters of the OCC model.

4.3 Features for Each Frame

Due to the fact that we want to build a generic framework for video segmentation without using specific knowledge about the target application scenarios and definition of

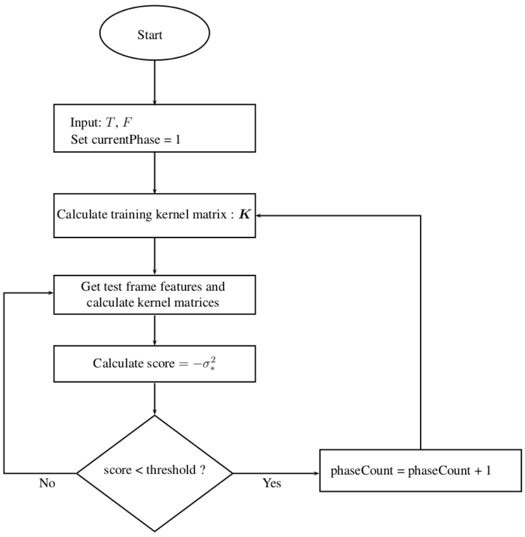
Start
Input: T, F
Set currentPhase = 1
Calculate training kernel matrix : $\boldsymbol{K}$
Get test frame features and calculate kernel matrices
Calculate score $= -\sigma_*^2$
score < threshold ?
No
Yes
phaseCount = phaseCount + 1

Fig. 1. Overview of our one-class classification approach for video segmentation

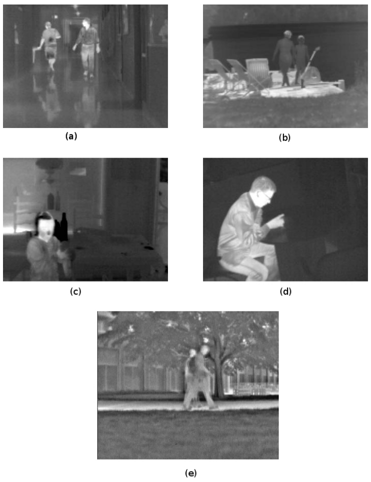

Fig. 2. Example frames of the five sequences within the CVPR change detection dataset: (a) *corridor*, (b) *lakeSide*, (c) *diningRoom*, (d) *library*, (e) *park*

events, it is extremely challenging to choose a suitable feature set for the algorithm. For the present implementation evaluated on the thermal surveillance videos, we use pyramidal histogram of oriented gradients (PHOG) as proposed in [1]. This is a very suitable feature because in surveillance sequences, events are defined by entry of a person, change in the normal movements of the people in the scene, etc., and PHOG features are very efficient at representing object shapes in the frame. In this feature representation, local features are represented as histograms of edge orientations. The number of bins is a parameter of the feature descriptor. Each histogram bin represents the number of edges orientated in a specific direction. To represent shapes of various sizes within a frame, an image pyramid is built for the frame and histograms from each pyramid level are concatenated. For more details, the reader is referred to [1].

5 Experiments

As generic semantic segmentation of video sequences is a new challenge in computer vision, we are not aware of any existing dataset specifically intended for this purpose. For this reason, the change detection dataset for the CVPR 2012 change detection workshop [2] is used to test our approach and to demonstrate its suitability. The thermal images from this dataset have been used because they contain sequences with large variance in object size and intensity contrast. Example frames of the dataset are shown in Figure 2. The ground truth data provided with the dataset is oriented towards motion detection and hence, we created our own ground truth for our application.[1] The data is in the form of grayscale frames of size 320x240 pixels. There are five sequences, namely *corridor, diningRoom, lakeSide, library, and park*. Each sequence contains different kind of motion and different zoom levels such that the object sizes are very different.

5.1 Experimental Setup

The parameter F, i.e., the number of frames used for training the model, was fixed at 50. Thus, the first 50 frames are assumed not to have any special event. At the normal rate of 25 frames per second, this means we are assuming constancy for only 2 seconds, which is a very realistic and reasonable assumption. The threshold T was set to 0.175, which was determined empirically on the dataset. Future work will include automatic determination of even these two system parameters. Furthermore, 30 bins are used in the PHOG descriptor for each histogram calculated over three levels.

5.2 Results on the CVPR Change Detection Dataset

In most of the published works including [5], the performance measures used are subjective and do not lend themselves to comparison. The reason is that in problems like video segmentation, it is very difficult to define a good performance measure for the algorithms. Hence, we concentrated on the fact that our algorithm works on the principle of detecting events and use performance measures for event detection. To evaluate the results of the algorithm qualitatively, we used the detection rate η:

$$\eta = \frac{\text{number of correct detections}}{\text{number of events in ground truth}} \quad . \tag{4}$$

It is often the case that the detection of the algorithm and the ground truth vary by about 20-25 frames, because the algorithm makes hard decisions using a threshold and ground truth is marked by human observers. This is not a serious problem, since in real-life videos 25 frames corresponds to a time span of 1 second, in which generally not many events happen. For most applications, this difference is not a major problem.

Additionally, over-segmentation is expected, because our algorithm works completely unsupervised. As the threshold is set without any prior knowledge about the video and is thus completely independent of it, often very small and insignificant changes in the video result in a new segment being reported. However, this is also not a cause for alarm as this stage is usually intended to be followed by a higher processing stage or a human observer in most applications. Thus, at the higher level, we can choose to ignore these particular extra segments in a post-processing step. We represent the effect of over-segmentation with the over-segmentation ratio γ:

$$\gamma = \frac{\text{number of false detections}}{\text{number of events in ground truth}} \quad . \tag{5}$$

[1] Readers interested in obtaining the ground truth for their own further research or verification of our methods can contact the authors.

Table 1. Results on the thermal video subset of the CVPR Change Detection Dataset for $F = 50$ and $T = 0.175$ using detection rate η (4) and over-segmentation ratio γ (5)

Video	η	γ
corridor	0.68	0.31
diningRoom	0.81	0.45
lakeSide	0.17	0
library	1	2.78
park	0.83	0.5

This represents the average number of extra segments for every segment in the ground truth. However, it should be noted that under-segmentation could be more costly and hence needs to be reduced. Table 1 shows the results for each video in the dataset.

The results for the *lakeSide* video are evidently poor compared to other videos. It is heavily under-segmented. This is due to the fact that the video has extremely low contrast and even for a human observer, it is very challenging to locate events. In addition, the events and objects in the video are of very small dimensions. To detect events in this scenario, one has to set the threshold extremely low, which would increase the false detection rate in other generic cases. On the other hand, we see that the *library* video is over-segmented. This can be solved by increasing the threshold (experiments revealed that increasing the threshold to 0.25 in this case reduces false detection rate to 0.35, while retaining the excellent detection rate).

Often, it is even desirable to have over-segmented videos, e.g. in the *library* video, the detected extra events are basically the person under observation turning pages. These are not labeled in the ground truth because they are minor events but could be interesting for the application. In the *diningRoom* video, the extra events detected are basically the person turning, which is an interesting change but again not labeled in the ground truth.

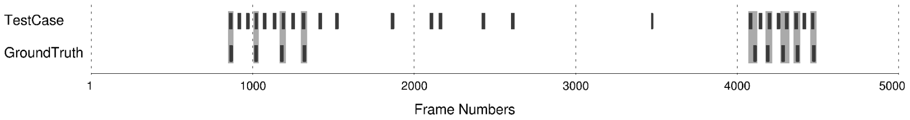

Fig. 3. Segmentation timeline for the *library* video (with $T = 0.175$ and $F = 50$), with the ground truth (the shadings indicate the matched detections)

Figures 3 and 4 show a timeline and an example frame segmentation on the *library* video, and Figure 5 shows segmentation example frames for *corridor* video. To analyze the effect of the parameter T on the overall performance, the η vs. γ curve is plotted for the *library* and *corridor* videos in Figures 6 and 7. We have varied T from 0.15 to 0.20.

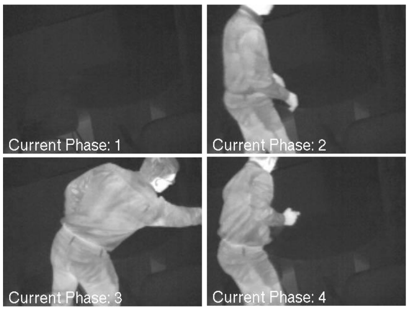

Fig. 4. Example segmentation on the *library* video

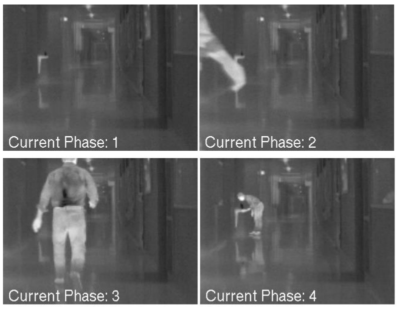

Fig. 5. Example segmentation on the *corridor* video

Larger values of the threshold result in an over-segmented video, i.e., high γ, whereas smaller values result in under-segmented videos. An interesting point to note is that for over-segmented videos, we obtain detection rates η often less than those with lower γ. This is due to the training time F. Often, an event is detected when there is none (i.e., over-segmentation) and during the following training time, an event is missed.

The results are promising and there are strong indications that a video dependent threshold determination will solve most of these existing issues.

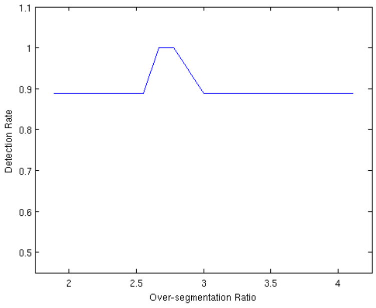

Fig. 6. Detection rate η vs. over-segmentation ratio γ for the *library* video

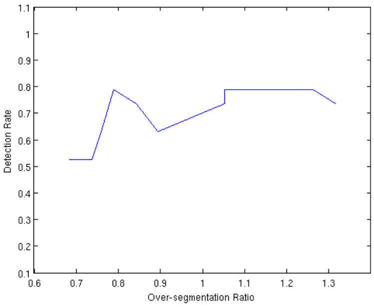

Fig. 7. Detection rate η vs. over-segmentation ratio γ for the *corridor* video

6 Conclusions and Future Work

The work was aimed at segmenting the video sequences into activity phases, enabling us to further process the smaller units and extract the interesting parts. We used a one-class classification approach based on Gaussian processes to do this with success as seen in the previous section. The accuracy in event detection is quite impressive even with the simple PHOG features used in this implementation. This demonstrates the possibility of using one-class classification schemes, such as the ones based on GPR, for the task of video segmentation.

As noted earlier, the accuracy of the system can be further enhanced by intelligent selection of parameters. Automated parameter optimization is one topic of future work. This may yield better results because then the parameters will be dependent on the video instead of being universal and it may avoid situations as the one encountered in the case of the *lakeSide* video.

Furthermore, feature selection will be part of further research. The authors believe that feature selection is of critical importance in this case and the use of more sophisticated features could drastically improve the performance of the approach. To exploit features representing shape and motion in a combined descriptor is a promising idea for future work.

7 Acknowledgements

Mahesh Venkata Krishna is funded by a grant from the Carl Zeiss AG, through the "Pro-Excellence" scholarship of the federal state of Thuringia, Germany. We thank Dr. Stefan Saur and Dr. Christian Wojek for useful discussions and suggestions.

References

1. Bosch, A., Zisserman, A., and Munoz, X. (2007). Representing shape with a spatial pyramid kernel. In *Proceedings of the 6th ACM international conference on Image and video retrieval (CIVR'07)*, pages 401–408, New York, NY, USA. ACM.
2. Goyette, N., Jodoin, P.-M., Porikli, F., Konrad, J., and Ishwar, P. (2012). changedetection.net: A new change detection benchmark dataset. In *Proceedings of the IEEE Workshop on Change Detection (CDW'12) at CVPR'12*.
3. Kemmler, M., Rodner, E., and Denzler, J. (2010). One-class classification with gaussian processes. In *Proceedings of the Asian Conference on Computer Vision (ACCV'10)*, pages 489–500.
4. Koprinska, I. and Carrato, S. (2001). Temporal video segmentation: A survey. *Signal Processing: Image Communication*, 16(5):477 – 500.
5. Liu, T., Zhang, H.-J., and Qi, F. (2003). A novel video key-frame-extraction algorithm based on perceived motion energy model. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(10):1006 – 1013.
6. Parzen, E. (1962). On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076.
7. Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.
8. Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., and Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471.
9. Tax, D. M. J. and Duin, R. P. W. (2004). Support vector data description. *Machine Learning*, 54(1):45–66.