Temporal Saliency in Videos

Nicholas Baker
Johns Hopkins University
Dept Cognitive Science, Johns Hopkins Univ, Baltimore, MD, USA
nbaker9@jhu.edu

Mentor: Nasim Souly
University of Central Florida
Center for Research in Computer Vision, University of Central Florida, Orlando, FL, USA
nasim.souly@gmai.com

Abstract

We discuss two experiments that deal with saliency in videos. In the first experiment, we combine motion detection algorithms with a context-aware saliency detection algorithm to add motion as a feature in video saliency. In the second experiment, we apply principles from spatial saliency in static images to pick out salient frames in a video.

1. Introduction

Faced with an enormity of information to process, humans and primates are able to quickly and easily locate and attend to semantically important regions in a visual scene. Before any top down, task oriented heuristics are used to guide eye movements, the primate visual system already begins to attend to areas of higher visual salience which, through knowledge-free analysis of the scene, guides their focus to areas of higher probable semantic importance.

Visual salience, as defined by Laurent Itti, is ''the distinct subjective perceptual quality which makes some items in the world stand out from their neighbors and immediately grab our attention'' [4]. In other words, the measure of saliency in a certain region of a visual scene is determined by its dissimilarity to other regions in the scene. Dissimilarity is determined by differences in a number of features, including intensity, color, and orientation. The more unique an area is across these features, the more salient it is said to be.

In computer vision, saliency has traditionally dealt with static images. Algorithms first began to be developed after Treisman and Gelade [9] proposed a model for feature integration, and shortly after, Koch and Ullman [7] developed saliency maps, or images whose luminance corresponds to the saliency of that particular region [1]. Since this early work, a tremendous volume of research has been carried out on the most precise and computationally efficient ways to compute the visual salience of a scene. In Itti's cognitive based approach [6], color, intensity and orientation are used as features, and variation across these three features is used to compute saliency. Goferman et. al used comparison of color and intensity, and were able to identify contextually salient elements of the background as well as the foreground by examining global and local feature contrasts [3].

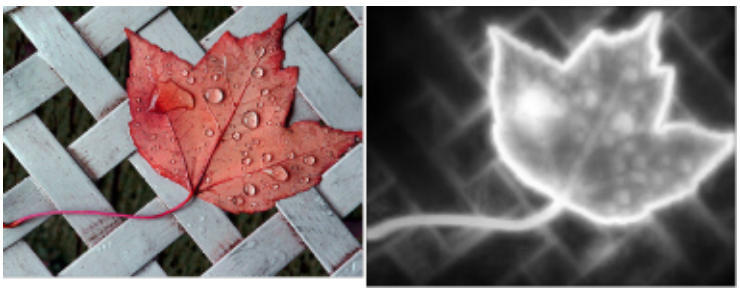


Figure 1. In Goferman's context-aware saliency algorithm, both the leaf and the mesh behind it are picked out as salient regions.

Saliency research has also extended to the domain videos, where motion plays an important role in determining salience. Things that are moving are more likely to attract a viewer's attention and be semantically relevant to a visual scene than stationary objects that may have greater feature contrasts in orientation, intensity and color. Xue et. al [10] captured this phenomenon in their algorithm, which picks out the most salient items by their motion. Itti et. al [5] computed surprisal using Shannon information, claiming that those locations in spatial and temporal space with the greatest surprisal represented the most salient regions of the video.

In our first experiment, we extended Goferman's

context-aware saliency algorithm into video by adding motion as a feature.

In our second experiment, we applied the notion of dissimilarity in intensity and color to the temporal dimension, seeking to pick out those frames in a video in which a salient event begins to occur. Just as novel regions in a static image are more likely to attract a viewer's attention and be semantically meaningful, novel frames in a video are likely to be the beginning of a semantically meaningful event.

2. Experiment 1

When we first attempted to handle saliency in videos, we aimed to extend Goferman's contextual saliency algorithm [3] to work for videos as well as static images. We aimed to construct a framework that would pick out the salient objects in the foreground of each frame, but also those components of the background that were instructive about the semantic content of the scene.

A naive approach to extending Goferman's algorithm would be to simply compute the contextual saliency of each frame and combine the outputs to create a saliency map for the video. However, this method proved inadequate, because the most salient objects in a static image may not be the most salient in a video due to the addition of motion as a relevant feature. Objects that are more salient in a single frame due to dissimilarity in color, orientation, or intensity might be less salient in the context of a video if they are stationary.

In order to account for this, we used motion detection algorithms to identify objects in motion, and increased the saliency values of those areas where motion was present.

2.1. Methods for Experiment 1

The first thing we tried to implement contextual saliency in videos was a late fusion between Xue's method of detecting motion saliency through low-rank sparse decomposition [10] and Goferman′s method [3] for static contextual saliency. First, we used Xue's algorithm to detect motion and record locations of motion in each frame. Next, we computed the contextual saliency map using Goferman's algorithm and boosted the saliency in the map where motion was detected using Xue's algorithm.

The method proved successful for increasing saliency values in areas where motion was present, compared to solely using the saliency map computed from Gofermans algorithm, while still preserving the background information that makes the method distinct. The process proved to be computationally intensive, however. Both algorithms we fused were computationally intensive operations, and their combination proved to be cumbersome to produce.

We determined that Xue's motion saliency framework computed more than was needed for the purposes of our method, so we decided to try using background subtraction and optical flow as two lower level motion detectors to fuse with Gofermans contextual saliency algorithm.

Although we tested with both background subtraction and optical flow, we ultimately decided that optical flow was the preferred method for motion detection in our framework. Optical flow can detect moving objects even in the context of a moving camera, which made our algorithm more robust and allowed us to test on a more diverse data set. Moreover, optical flow gave us more information about the speed and direction of motion of objects in the video, which allowed us to perform graded modifications on the saliency map depending on the magnitude of motion.

2.2. Results for Experiment 1

We tested our framework using optical flow and Goferman's context-aware saliency algorithm, with videos from Institut fuer Neuro- und Bioinformatik [2]. Quantitative assessment of the videos proved difficult because of the somewhat ambiguous goal of contextual saliency. In traditional saliency experiments, the goal is to accurately predict eye movements of human subjects who are presented with the same image or video that the algorithm's saliency map is computed from. If the saliency map's highest valued areas are also the areas most likely to attract the gaze of human subjects, the algorithm is successful. However, contextual saliency's aim is to capture the semantically relevant regions of a visual scene, whether or not they correspond to human gaze data.

In practice, since the purpose of vision is to understand the semantic content of a scene, the goals of human gaze and contextual saliency are highly correlated. Even if a region of the scene is not an object in the foreground, if it is necessary to understanding the semantic content of the scene, it should still attract the gaze of human subjects. But the evaluation the saliency map for contextual saliency remains qualitative: does the map produced accurately select the most semantically relevant elements of each frame of the video?

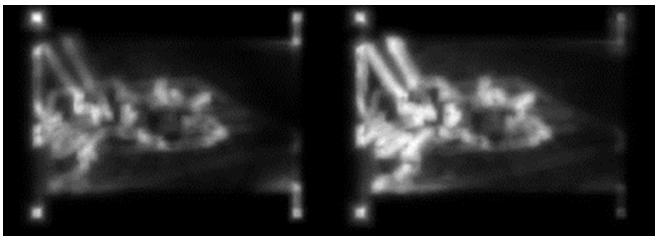


Figure 2. Saliency maps from a video of a man placing a puppy on a mat with other puppies. Goferman's method is on the left, and our fused method is on the right. Notice that areas of motion (the man's arm and the puppies on the mat) are brighter on the left.

Our results indicated that the framework was generally successful at accounting for motion in context-aware

saliency. The overall performance was comparable to Goferman's static image contextual saliency algorithm, but with the added effect of picking out as salient those objects that were in motion, even if their dissimilarity across the other three features was relatively small.

2.3. Experiment 1 Conclusions

Our method of fusing optical flow and Goferman's contextual saliency algorithm was successful in extending the contextual saliency method to include motion as a feature, making it perform well for videos. Qualitatively, those areas of semantic importance were picked out using the framework we constructed. Regions that were in motion or were dissimilar by color or intensity were picked out as salient, and regions that were both dissimilar and in motion were picked out as especially salient.

Though the algorithm functioned, it was a somewhat inelegant coupling of two existing algorithms. More robust methods of computing saliency in videos already exist [5, 8], though none that included Gofermans work on the issue of context.

We wanted to continue working on saliency as it applies to videos and global dissimilarity, but to develop our own system using what we had learned from our work with existing methods. With this goal in mind, we began work on Experiment 2.

3. Experiment 2

In Experiment 2, we set the new goal of locating frames, rather than regions of frames, that were dissimilar to the frames posterior to them. The idea was to take what is known about saliency in the spatial dimension and apply it to the temporal dimension. In this way, we hoped to isolate frames where a semantically new event was beginning in video clips.

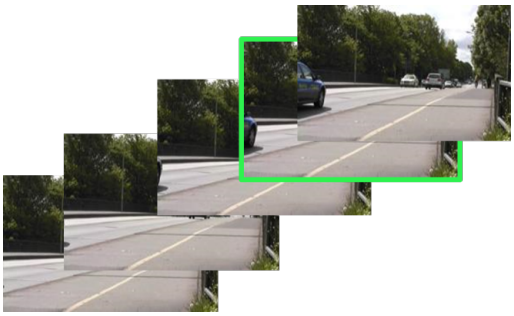


Figure 3. Disimilarity between a frame and all prior frames was calculated to pick out the onset of a new event (outlined in green).

3.1. Overview of Methods from Experiment 2

Input was provided from videos, whose frames were then resized so that the largest dimension is 250 pixels. Each frame was divided into 7x7 pixel patches, with 50 percent overlap, per Goferman's method.

To compute the saliency of each frame, we measured the dissimilarity between each patch in the frame being considered with the analogous patch in all frames before it. We summed the dissimilarity of all patches in the image to get one value for the frame's overall dissimilarity.

3.2. Dissimilarity

Dissimilarity between a patch in image a and the analogous patch in a previous image b was calculated by the Euclidean color distance between the two patches.

$$FrameDif(a,b) = \sum_{i=1}^{w} d_{euclidean}(a_i, b_i) \qquad (1)$$

Where w is the total number of patches in either frame a or frame b.

In order to calculate the frame's total dissimilarity, we summed the dissimilarity of the frame across the total number of frames before the frame in question, the f^{th} frame. We then divided by the total dissimilarity by f.

$$Dissim(frame_f) = \frac{\sum_{k=1}^{f} FrameDif(frame_f, frame_k)}{f} \qquad (2)$$

3.3. Temporal Distance Weighting

Our algorithm intends to identify frames at which a novel event begins to occur, so frames that were closer to the current frame of consideration were weighted more heavily than frames that occurred much earlier in the video. To capture this, we added a decay factor to the frames dissimilarity computation. As frames' temporal distance from the frame of consideration increased, the impact of their dissimilarity on the total dissimilarity computation for the frame decreased.

We incorporated this into our model by multiplying the dissimilarity of the frames with the ratio of the number of frames preceding the comparison frame and the number of frames preceding the frame of consideration.

$$Dissim(f) = \frac{\sum_{k=1}^{f} (FrameDif(frame_f, frame_k) * \frac{k}{f})}{f} \qquad (3)$$

3.4. Grouping

The onset of an event would often produce a series of rapid increases in dissimilarity before adaptation occurred. In order to account for this, we suppressed events above the threshold if they occurred in close proximity to an earlier event.

In our algorithm, if a frame exceeded the threshold but was within four frames of another frame that exceeded the threshold, the later frame was suppressed and ignored. This way, the algorithm picked out only those frames which represented the beginning of an event and ignored frames with little semantic interest despite a potentially sizable physical difference between frames.

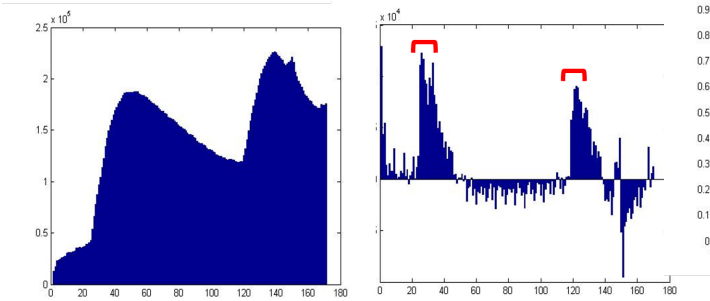


Figure 4. After taking the derivative of the curve, we grouped nearby areas of high activation by their first frame

3.5. Results for Experiment 2

The purpose of our algorithm was to pick out the beginning of events in videos. To test properly, we needed to identify videos that fit a specific set of criteria. Good test videos had a clear starting event some time into the clip. We wanted to determine if the algorithm could successfully pick out this new event or action. We used surveillance videos from Youtube and a test video from Institut fuer Neuro- und Bioinformatik [2] to assess the algorithms performance.

As a pilot trial, we created a synthetic video where a bright red square appeared in one of the frames while a woman ran across the screen. Although there were changes between every frame because of the womans movement, the system picked out the frame with the red square as the important frame in the video clip.

We then began testing on natural videos to see if semantically new events were captured by our algorithm. The first video was of a street with heavy traffic. Though there is constant motion and activity during the video clip, the system picked out only those events where a new car was introduced to the field of view. Activation from a new object decreased as the system acclimated to its presence. We tested on a clip where one new car drove into the scene, and then tried a longer video in which two cars drive in sequentially. The algorithm picked out both cars' entrance into the cameras view.

We also tested on surveillance videos from Youtube. One surveillance video was taken from behind the counter at a convenience store. The algorithm unexpectedly activated at a part of the clip where no event was present. A second, smaller activation also took place at the ground truth event.

The second surveillance video we tested was a security camera at a gated entrance. An event was triggered before the truck moved into view of the camera. Its headlights illuminated part of the scene, which created enough dissimilarity to signal a new event in the system. Activation continued to increase when the truck pulled into view of the camera.

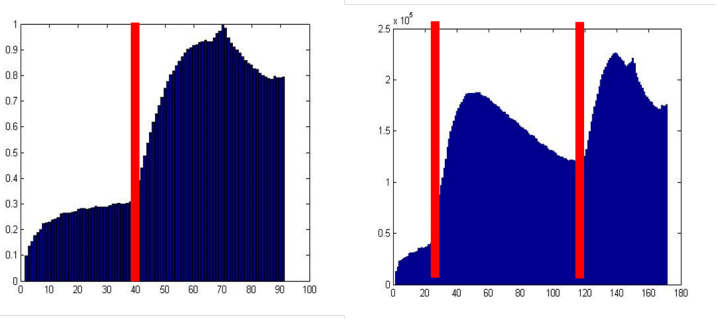


Figure 5. In the first clip (left), one car drove into the street camera. In the second, two cars drove into the field of view. The red line denotes the ground truth of when the event began.

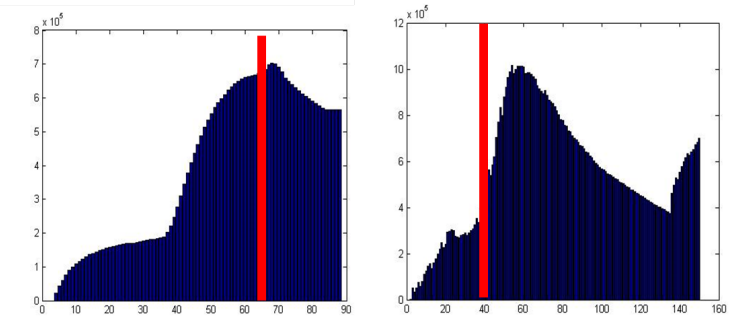


Figure 6. The algorithm did not as accurately pick out the ground truth for the surveillance video as from the street video clips.

3.6. Experiment 2 Conclusions

For videos ideally suited for its methods, the algorithm developed in Experiment 2 is overall fairly effective. It succeeds in picking out the beginning of a novel event within one or two frames of the ground truth we had determined prior to testing each video clip, and plausibly acclimates to events after a few frames, decreasing in activation until another event occurs.

The nature of the algorithm's extremely low level, bottom-up approach somewhat constrains the breadth of its capabilities. Though we hoped to pick out those frames where a meaningful new event occurred, the algorithm has no concept of what is meaningful or not meaningful apart from what produces the greatest dissimilarity with previous scenes. A car driving through the scene, for example, would be picked out with much greater certainty than a person pointing a gun at another person in the scene. Such top-down computations require action and object recognition. These high level processes are outside the scope of

this algorithm, which aims to pick out events with only the most low level data available in the videos.

4. Overall Conclusions and Future Directions

In Experiment 1, we explored the role of motion in saliency of videos. We succeeded in adjusting Gofermans algorithm to capture the importance of moving objects in videos, while preserving its performance in selecting those objects and regions of the background that were most salient as static images.

In Experiment 2, we leveraged the work done on saliency in 2d image space to explore saliency in the one dimensional temporal space. We were able to pick out those frames that were most dissimilar to the frames preceding them, under the assumption that those frames of greatest dissimilarity would represent the onset of a new event in the video clip. This assumption was validated in our testing, as the frames picked out by our algorithm matched those picked out a priori by hand.

The work we have done in Experiment 2 focuses solely on the temporal dimension. In the future, it would be instructive to extend the algorithm to the spatiotemporal dimension, picking out not just the frames that were dissimilar, but the regions within the frames that caused this dissimilarity.

5. Acknowledgments

We thank the NSF whose funding made this challenging and engaging experience possible. I would also like to extend my heartfelt gratitude to Dr. Mubarak Shah and Dr. Niels da Vitoria Lobo, whose continuous support, advice, and encouragement kept me motivated and on track throughout the duration of this research project.

References

[1] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE*, 35(1):185 – 207, 2013.

[2] M. Dorr, T. Martinetz, K. Gegenfurtner, and E. Barth. Variability of eye movements when viewing dynamic natural scenes. . *Journal of Vision*, 10(10):1 – 17, 2010.

[3] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 34(10):1915 – 1926, 2012.

[4] L. Itti. Visual salience. 2(9):3327, 2007.

[5] L. Itti and P. Baldie. A principled approach to detecting surprising events in video. *CVPR*, 2005.

[6] L. Itti, C. Koch, and E. Neibur. A model of saliency-based visual attention for rapid scene selection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(11):1254 – 1259, 1998.

[7] C. Koch and S. Ullman. Shifts in selective visual attention: Towards the underlying neural circuitry. *Human Neurobiology*, 4(4):219 – 227, 1985.

[8] N. Souly and M. Shah. Context-aware saliency detection. *In Progress*, 2013.

[9] A. Treisman and G. Gelade. Modeling the influence of task on attention. *Vision Research*, 45(2):205 – 231, 2005.

[10] Y. Xue, X. Guo, and X. Caoi. Motion saliency detection using low-rank and spare decomposition. pages 1485 – 1488, 2012.