SALIENCY DETECTION IN VIDEO

Jonathan Wu

5/20/2012

Boston University

Department of Electrical and Computer Engineering

Technical Report No. ECE-2012-03

BOSTON UNIVERSITY

SALIENCY DETECTION IN VIDEO

Jonathan Wu

Boston University
Department of Electrical and Computer Engineering
8 Saint Mary's Street
Boston, MA 02215
www.bu.edu/ece

5/20/2012

Technical Report No. ECE-2012-03

Contents

List of Figures

1 Introduction

Video has become prevalent in modern society, with years of content accessible through the web. With such widespread availability, tools to monitor these videos become necessary to prevent malicious behavior (copyright infringement, inappropriate materials), or in live video, protecting safety. The focus of this work is aimed towards a subset of these videos - primarily those dealing with surveillance.

In surveillance applications, it would be useful to automatically annotate abnormal behavior in video. In many cases, a handful of people need to monitor hundreds of monitors (Fig. 1). Practically, this is inefficient, as it is hard to focus on so many cameras at once. Thus, an automated system (that lacks this limitation) could be built to be faster and more responsive than any operator.

Figure 1: Modern department of transportation traffic monitoring rooms

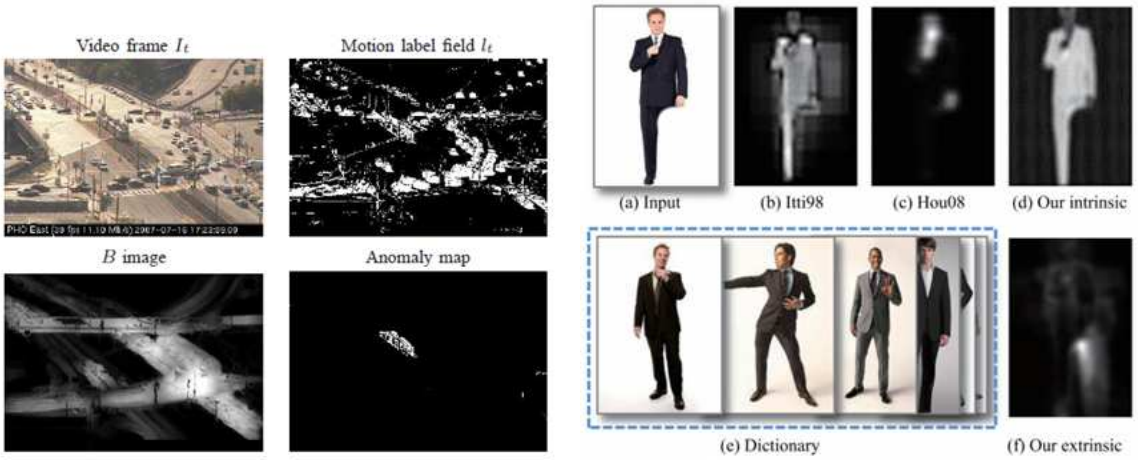
The general challenge of this work is to determine what is abnormal or suspicious. Abnormalities can generally be put into two categories: objects and behaviors. Objects are abnormal by having different photo-metric properties (looks different). Behaviors are abnormal by having different motion (moves differently). The focus of this work is the latter, i.e., determining events of note (saliency) through varying motion.

It is important to note that these abnormal behaviors are generally not known beforehand - only normal behavior is known. Thus, this can become hard to solve.

2 Related Work

Techniques involving object recognition, action recognition, tracking, and behavior subtraction[1] have all been proposed as ways to detect abnormality. Object and action recognition techniques identify anomalies by noting deviations from normally detected objects and actions. However, these techniques are vulnerable to occlusions and high computational costs (due to feature extraction, and multiple objects in a scene). Behavior subtraction uses background information to calculate changes in

scene dynamics (changes from the foreground) to determine abnormalities. Wang et al.[6] uses block-based image warping and k-nearest-neighbor (KNN) matching from a dictionary to determine still image saliency.



(a) Behavior subtraction [1] (b) Image saliency [6]

Figure 2: Results of related works

The goal of this work is aimed at expanding the last approach to video using similar ideas but applied to motion vectors.

3 Problem Statement

The goal of this project is to annotate salient events (defined earlier as abnormalities in motion) in any particular video sequence. Salient events are shown as a binary black/white mask, where white denotes an abnormality in the sequence. An overview of the method is presented in Figure 3.

The method presented in this report uses KNN optical flow matching of 3D blocks to generate a saliency map. This map is then segmented using Markov Random Fields (MRF). The resulting saliency mask is evaluated against a known ground truth to generate receiver operating characteristic (ROC) curves using various MRF parameters. These steps are described in more detail in later sections.

3.1 Input Sequences

To generate a saliency block, a sequence containing non-salient information has to be known. For a generic sequence, we can buffer the initial frames of a sequence and use the initial period as our dictionary (assuming it does not contain anomalies). The rest of the sequence can be the "query" sequence. For the dataset used in this work, the training and testing "query" sets were given - with training sets containing no anomalies.

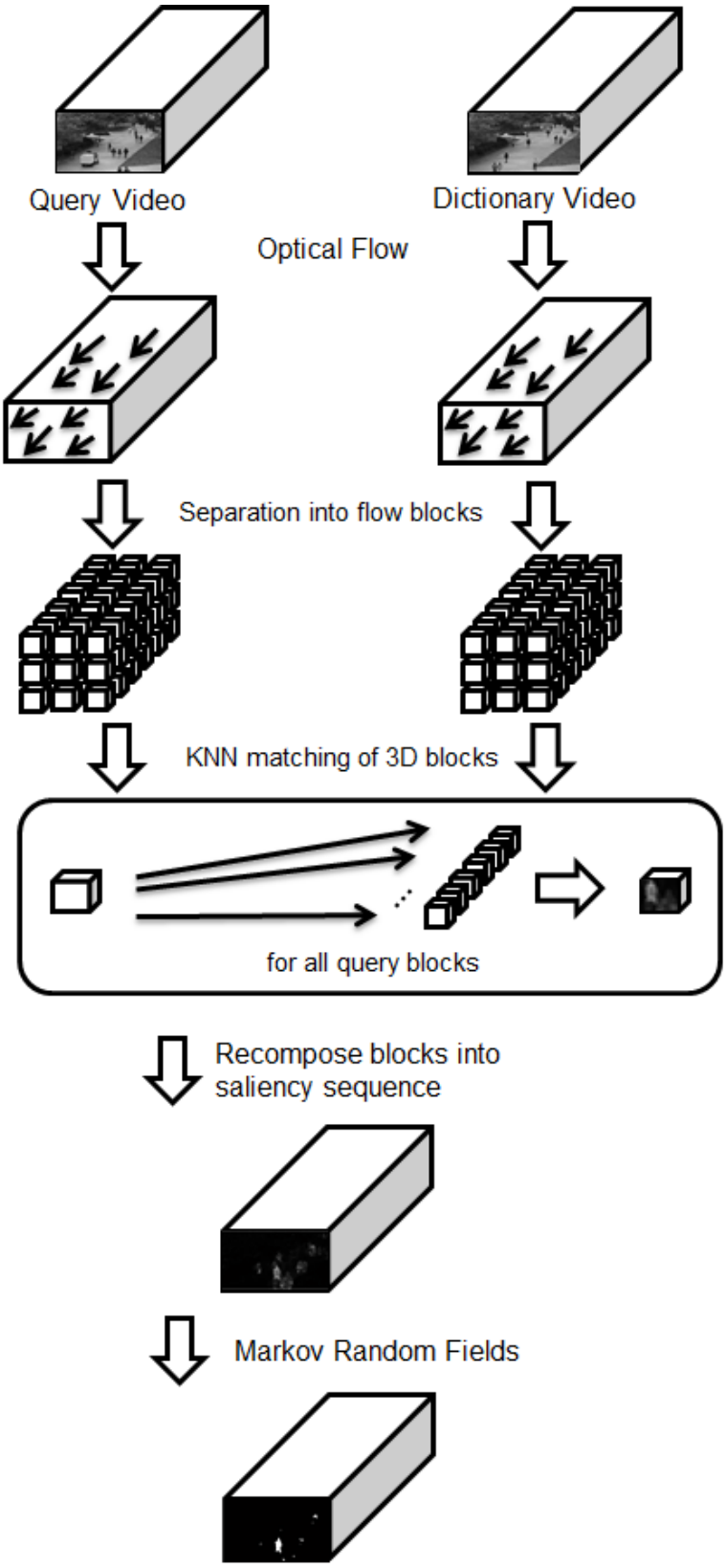


Figure 3: Overview of saliency detection method

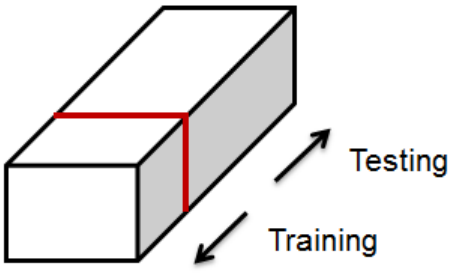


Figure 4: Splitting a generic sequence into training and testing parts using buffering

3.2 Dense Optical Flow

Optical flow can be computed using the Lucas-Kanade method[3] on every pixel in the sequence. Generally, this is the largest computational sink in the method and is pre-computed across all the sequences. This method is described below.

Following the image constraint equation, let I denote a grayscale video frame :

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$
$$\frac{dI}{dx}V_x + \frac{dI}{dy}V_y + \frac{dI}{dt} = 0 \text{ (Taylor series expansion)}$$

where $I_x = \frac{dI}{dx}$, $I_y = \frac{dI}{dy}$, $I_t = \frac{dI}{dt}$ are the directional partial derivatives and V_x and V_y are the motion vectors that need to be found.

Lucas Kanade assumes that motion has small displacements, and that the nearby neighborhood of a pixel has similar motion. Using this assumption, windows of nearby pixels are taken (assumed to have the same motion vectors), and solved using linear least squares for the image constraint equation.

$$A = \begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_n) & I_y(p_n) \end{bmatrix}, \vec{v} = \begin{bmatrix} V_x \\ V_y \end{bmatrix}, \vec{b} = \begin{bmatrix} -I_t(p_1) \\ -I_t(p_2) \\ \vdots \\ -I_t(p_n) \end{bmatrix}$$

$A\vec{v} = \vec{b}$ cannot be satisfied exactly so a linear least squares solution is used.

$$A^T A\vec{v} = A^T\vec{b}$$
$$v = (A^T A)^{-1}A^T\vec{b}$$

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum_i I_x(p_i)^2 & \sum_i I_x(p_i)I_y(p_i) \\ \sum_i I_x(p_i)I_y(p_i) & \sum_i I_y(p_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i I_x(p_i)I_t(p_i) \\ -\sum_i I_y(p_i)I_t(p_i) \end{bmatrix}$$

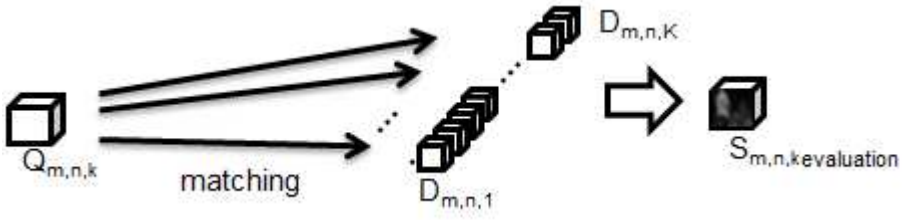


Figure 5: A $16x16x12$ query block is compared against to its closest K blocks. Compared blocks share the same spatial location, where $Q_{m,n,k}$ and $D_{m,n,K}$ denote the separated block indicies.

3.3 K-nearest-neighbors of matching 3D blocks

Every 3D optical flow block (designated query block) from the testing sequence is compared using KNN to 3D optical flow blocks from the training sequence. Blocks are only compared to those that share its same spatial location - thus, every spatial block location can be thought to have its own dictionary that consists of blocks from along its time axis. Furthermore, events that are not frequent in a particular spatial location can become anomalous. This is done so that events that are frequent in one spatial location, such as a car on a road would become anomalous if it were to appear in a nearby spatial location such as a sidewalk. K = 10 blocks was used in this project.

The metrics used for KNN are as follows:

$$\text{distance between two blocks} = \sum\nolimits_{x,y,t} \|\vec{Q}_{m,n,k}(x,y,t) - \vec{D}_{m,n,l}(x,y,t)\|$$

where $\vec{Q}_{m,n,k}(x,y,t)$ and $\vec{D}_{m,n,l}(x,y,t)$, denote the motion vectors at pixel location x,y,t in an indexed query and dictionary block. The indicies for $\vec{Q}_{m,n,k}$ and $\vec{D}_{m,n,l}$ denote blocks with the same spatial locations (m,n) in the sequence.

Once the KNN blocks were found, a saliency map is produced for each block (at every pixel location) as follows:

$$S_{m,n,k}(x,y,t) = \frac{1}{K}\sum\nolimits_{k=1}^{K} \|\vec{Q}_{m,n,k}(x,y,t) - \vec{D}_{m,n,k}(x,y,t)\|$$

where $S_{m,n,k}(x,y,t)$ denotes the saliency value at pixel location x,y,t in the block S.

This step yields a video sequence with a real-valued scalar saliency map.

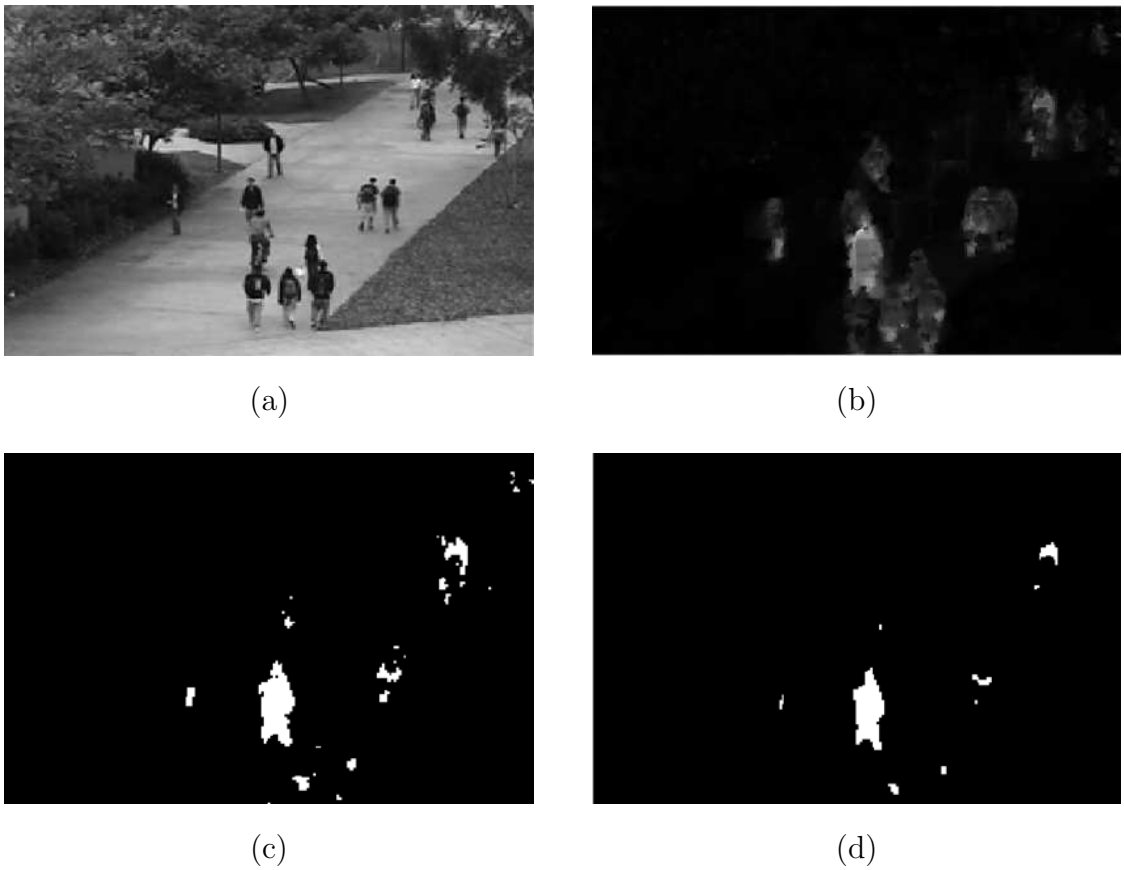
3.4 Markov Random Fields

Markov modeling was applied to segment salient regions from the generated saliency map. The model used incorporates second-order spatial neighborhood information in its Gibbs model to yield the following binary hypothesis test:

$$\Psi_k^2[x] \underset{NotSalient}{\overset{Salient}{\gtrless}} 2\sigma_{NS} ln(\theta \frac{\sigma_S}{\sigma_{NS}}) + \frac{N_{NS} - N_S}{T}$$

where $\Psi_k[x]$ is an observation from the saliency map modeled as a Gaussian random variable with standard deviation σ_S in salient regions, and standard deviation σ_{NS} in non-salient regions. T is the natural temperature of the Gibbs distribution, and N_{NS} and N_S are the number of non-salient/salient labels in the neighborhood.

This method is described in more detail in[4]. A few of the results are shown below in Figure 6.



(a) (b)

(c) (d)

Figure 6: Results of MRF: (a) Original video frame, (b) Corresponding Saliency Map, (c) Fixed threshold, (d) Variable threshold MRF segmentation

3.5 Assumptions

A few assumptions were made beforehand about the proposed method. The first is that the video sequence is from a static camera (no pan-tilt-zoom). This assumes that blocks with the same spatial location are related (that they are at the same location throughout the video). If the camera was moving, global motion would need to be calculated, and corresponding blocks would have to be warped (if the planar

view of the scene changed) and compared with their predicted spatial locations. The second assumption is that spatial scale is constant throughout. If this assumption is broken, motion vectors of an object towards the horizon are smaller and become more sensitive to the selected detection threshold. With additional scene information (type of scene/view) different thresholding could be applied throughout the sequence to improve results. Furthermore, the K-Nearest Neighbors search assumes that the training set is rich and diverse. This way, if an event were salient, it would find very few common neighbors yielding a high saliency value, and if it were normal, a very low saliency value would result.

4 Implementation

This method was implemented in MATLAB, using Liu's[2] optical flow library to compute motion vectors. Computation was done a desktop with an i7-2600K CPU @ 3.40 GHz with 16GB of DRAM.

4.1 Dataset

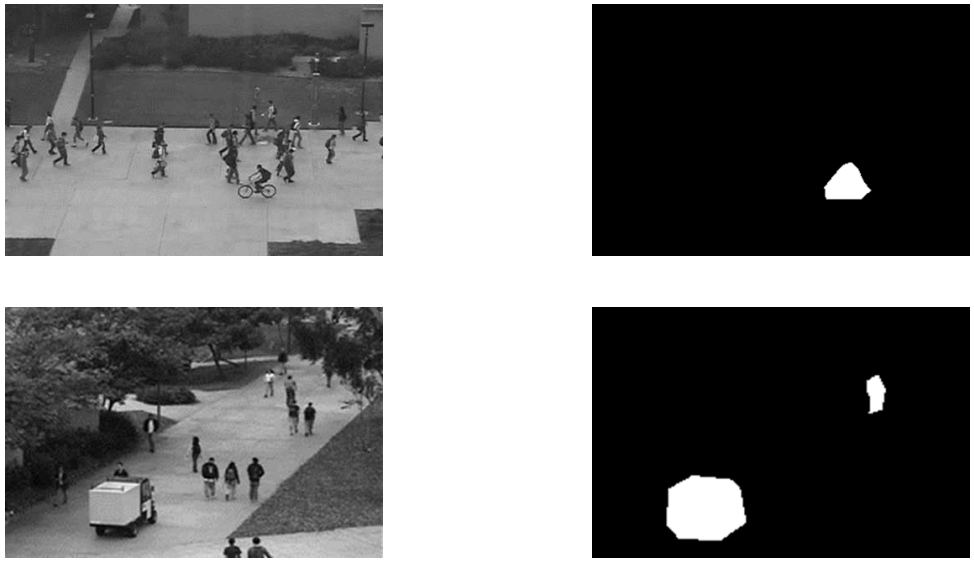
The UCSD Anomaly Dataset[5], contains video sequences captured by two fixed cameras overlooking a sidewalk. The dataset provides training sequences (no salient events) and testing sequences (with salient events). Furthermore, the testing set provides a few sequences with ground truths (binary masks) which mark out anomalous events. Typical anomalous events are due to unusual motion coming from vehicles, skateboarders, the handicapped, and cyclists. Figure 7 shows some sample images from the dataset.

5 Results

The novel method proposed above results in decent saliency detections (as shown in Figure 8). When evaluated as an ROC curve (in Figure 9) the 2nd order MRF has improved results over purely thresholded saliency. However, these results were obtained when assumptions are satisfied. In scenes where assumptions in Section 3.5 do not hold, results tend to fair much worse. For example, when motion scale is not spatially constant, motion that gets smaller due to perspective may not be detected properly.

6 Conclusions and Future Work

When assumptions hold, the proposed method performs fairly well. However, for future work, it may be useful to be able to relax some of these assumptions. As this model only incorporates motion, photometric information is not used. This allows visually abnormal objects to be potentially detected as normal behavior if



(a) Sample test video frames (b) Ground truths

Figure 7: UCSD dataset – anomalies are the vehicle and cyclists

their motion is similar. An extension to this work may find it useful to include additional photometric, texture, and edge information into the model. The static camera assumption can also be removed in future work if a global motion model is included as well. Additional work into the distance metric can be looked into to improve saliency detection while being robust to scene scale.

References

[1] P.-M. Jodoin, V. Saligrama, and J. Konrad, "Behavior subtraction," *Proc. SPIE Visual Communications and Image Process*, pp. 10.1–10.12, Jan 2008.

[2] C. Liu, *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*. PhD thesis, Massachusetts Institute of Technology, May 2009.

[3] B. D. Lucas, *Generalized Image Matching by the Method of Differences*. PhD thesis, Robotics Institute, Carnegie Mellon University, July 1984.

[4] J. M. McHugh, J. Konrad, V. Saligrama, and P.-M. Jodoin, "Foreground-adaptive background subtraction," *IEEE SIGNAL PROC LETTERS*, 2009.

[5] Statistical Visual Computing Lab, "UCSD anomaly detection dataset," 2011. http://www.svcl.ucsd.edu/projects/anomaly/dataset.html.

[6] M. Wang, J. Konrad, P. Ishwar, K. Jing, and H. Rowley, "Image saliency: From intrinsic to extrinsic context," *CVPR*, July 2011.

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

Figure 8: Results of the proposed method across various sequences. (a) Original Frame, (b) Saliency Map, (c) Segmented Saliency Map, (d) Ground Truths Map

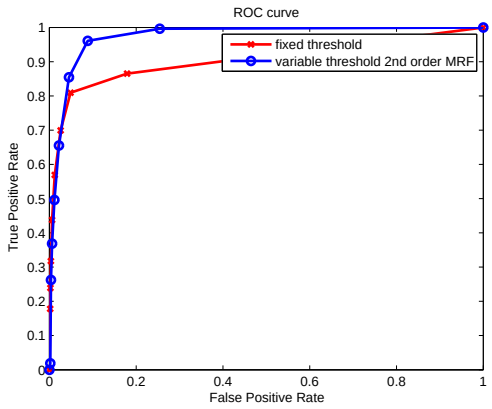


Figure 9: ROC curve