SEMANTIC DESCRIPTION IN STEREO VIDEO CONTENT FOR SURVEILLANCE APPLICATIONS

Nikos Papanikoloudis, Sotirios Delis, Nikos Nikolaidis, Ioannis Pitas
Department of Informatics
Aristotle University of Thessaloniki
Box 451, Thessaloniki 54124, GREECE
tel: +30 2310 996361
{nikolaid, pitas}@aiia.csd.auth.gr

ABSTRACT

The use of a stereo camera in surveillance applications adds information about the in depth position of the object being monitored. Thus, stereo videos can help extract semantic information about a person or object movement direction, not only on the image plane, but also in depth space. This paper describes a method that performs semantic labeling of movement direction in stereo videos along the horizontal and vertical axes and also along the depth axis when disparity information is available. A method that that extracts information about whether two or more objects are approaching or moving away is also presented

Index Terms — Semantic labeling, stereo video.

1. INTRODUCTION

Semantic annotation of object or human movement direction with labels such as left/right movement can lead to more efficient surveillance in indoor or outdoor areas. In addition, the integration of disparity information that can be derived from a stereo video, helps us to semantically label movement in the depth space. If more than one objects have been detected, we can also extract information about whether these objects approach each other or are moving away. Such a semantic annotation is also useful for stereoscopic video description for archival and retrieval purposes. In this paper, we propose methods for extracting such semantic information from stereo videos in the x,y plane and also in the depth space when disparity information is available.

The rest of this paper is organized as follows. In section 2 the algorithms that perform horizontal, vertical and along the depth axis movement characterization are described and a representative example for every algorithm performance is presented. In section 3 the algorithm that extracts information about whether two or more objects are approaching or moving away is described along with representative examples. Finally, conclusions are drawn in section 4.

2. MOVEMENT CHARACTERIZATION

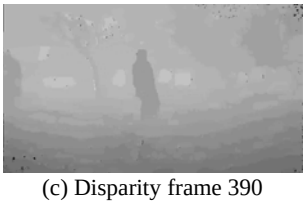
In this section, we present a method for characterizing movement in the image plane and in depth space for objects captured in a stereoscopic video for which disparity information is available. More specifically, the method applies labels to regions of interest (ROIs) that correspond to tracked objects or persons, for each one of the three axes x, y and d (depth). ROIs are first tracked in the left and right video channels. An example of a tracked ROI is depicted in Figure 1. Since no calibration data are available in most cases, we cannot map image coordinates and disparities (x, y, d) to global coordinates (X, Y, Z). Thus, we label object movement trajectory independently along the spatial x, y axes and along the depth axis.

2.1. Horizontal movement

In order to label the object movement along the horizontal axis, we use the x coordinate $x(i)$ of the center $c(x, y)$ of the object ROI over several video frames $i = 1,2,...,t$ as input. Then we smooth $x(i)$ by applying a simple mean filter of appropriate size.

(a) Frame 390

(b) Frame 880



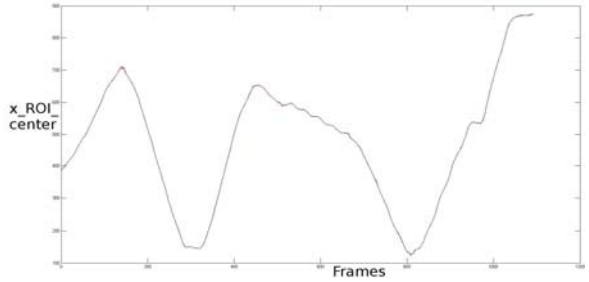
(c) Disparity frame 390

(d) Disparity frame 880

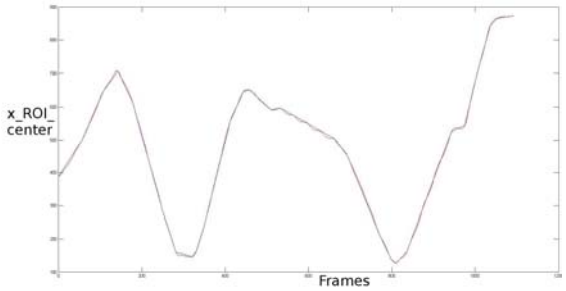
Figure 1. Sample video frames. The green marked ROI rectangles are the output of a tracking algorithm [7] . Disparity maps were extracted using algorithm [5] that is part of the OpenCV 2.3 [6] .

Finally we approximate the filtered signal using the linear piecewise approximation method [1] . The slope sign of a linear segment indicates whether the object is moving to the right or to the left direction. Figure 2a shows the filtered

signal $x'(i)$, $i=1,2,...,t$ where i is the frame number. Smoothing is performed using a mean filter of size 7. In Figure 2b the filtered signal is approximated by line segments using the linear piecewise approximation method. After approximation, horizontal object motion is semantically described by a list of $[t_l, s_l, a_l]^T$, $l=1,2,...,n$, where t_l, s_l, a_l are the start video frame, the duration and the slope of a line segment respectively and n is the total number of segments. Too short line segments are discarded. The slope sign defines the horizontal motion type "left movement", "right movement" and "still" indicated by a negative, positive or close to zero slope respectively.



(a)



(b)

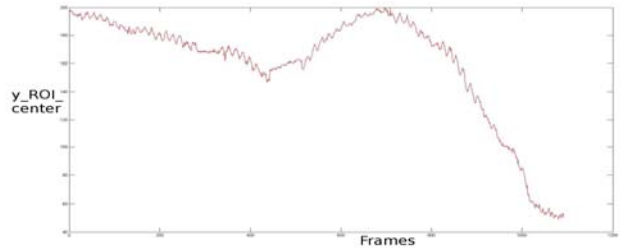
Figure 2. (a) The filtered ROI center x coordinate signal used as input to the linear piecewise approximation method, (b) the filtered and approximated signal.

The results label list for horizontal movement for the video shown in Figure 1 is: segments {1, 142}, {320, 442}, {809,942} and {974,1051} are categorized "right movement", segments {142,281}, {457,509} and {662, 809} are categorized "left movement" and segments {281, 320}, {442,457}, {509,662}, {942,974} and {1051,1092} are categorized "still".

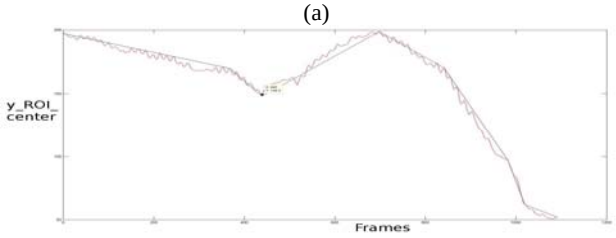
In case of a stereo video, such a semantic motion characterization can be performed on both stereo channels. Then, the resulting semantic descriptions are merged to a single description.

2.2. Vertical movement

To estimate the vertical movement direction, we follow a similar approach to the one described in the previous section. The difference in this case is that the input signal is the y coordinate $y(i)$ of the ROI center $c(x,y)$. In Figure 3 the raw and the filtered and linearly approximated input signal for the vertical motion direction characterization algorithm are depicted.



(a)



(b)

Figure 3. (a) The signal used as input, (b) the filtered signal and (c) the result of the linear approximation.

The results of the vertical movement direction for the video in Figure 1 are: segments {1, 366}, {700, 842} are categorized "still", segments {371, 438}, {698, 1020} are categorized "up movement" and segments {438, 698} are categorized "down movement".

2.3. Movement along the depth axis

It is well known that disparity is indicative of the distance of a point in real world from the camera center. Thus it can be used for in-depth movement direction estimation of a ROI. The algorithm first performs a pixel trimming technique [2] to remove background pixels from the object ROI in the disparity channel. Pixel trimming first computes the mean disparity d_m using all pixels inside a central region of the ROI. The region within which we compute the mean disparity in regard to the whole ROI is shown in Figure 4 (red rectangle).

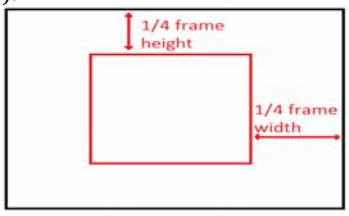


Figure 4. The region of the ROI used to compute the mean disparity of the ROI.

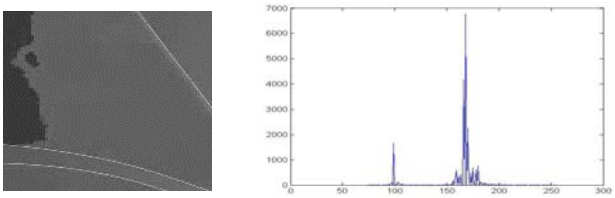


Figure 5. The disparity map of a ROI (a) and its histogram (b).

In order for a pixel within the ROI to be accepted as belonging to the object of interest, its disparity value $d(q,r)$ must be in the range [d_m -T, d_m +T], where T is an appropriately chosen threshold. Figure 5 shows the disparity map of a face ROI that has been tracked by a tracking algorithm. Figure 6 shows the histogram shown in Figure 5 after performing the pixel trimming technique. The background disparities have been efficiently trimmed out.

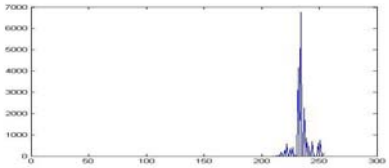
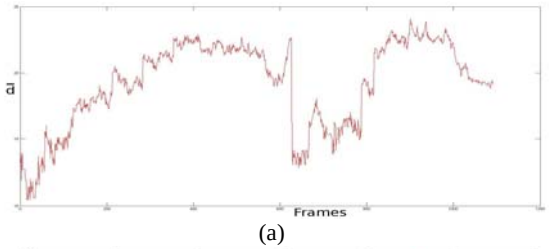
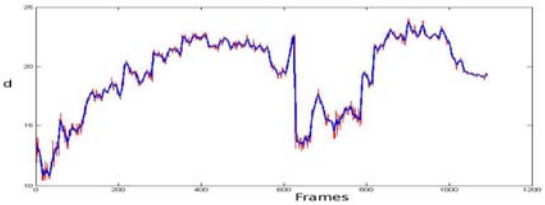


Figure 6. object disparity within the object after trimming.

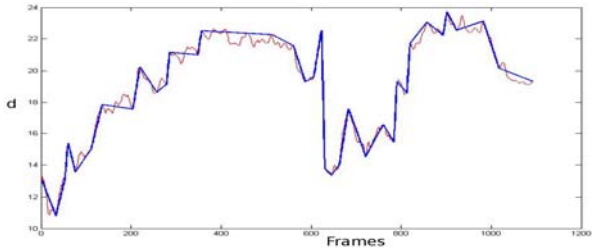
Then, we compute the mean disparity value of the remaining pixels $\bar{d}(i)$, i=1,2,...,t over all frames where the object has been tracked, perform smoothing of $\bar{d}(i)$ by a simple mean filter and approximate the filtered signal using linear piecewise approximation. Finally, the slope sign of the derived linear segments is used to characterize the object movement direction along the depth axis. Figure 7 shows the raw and filtered and approximated signal for movement direction estimation along the depth axis of the video shown in Figure 1.



(a)



(b)



(c)

Figure 7. (a) The signal used as input, (b) the filtered signal and (c) the result of the linear approximation.

The results for the movement direction in the depth axis are semantically described by a list of $[t_l, s_l, a_l]^T$, $l=1,2,...,n$, where t_l, s_l, a_l are the start video frame, the duration and the slope of a line segment and n is the total number of segments. Labels "back movement", "front movement", "still" are assigned to segments when a_l is negative, positive or close to zero respectively. An example of such a list for the video shown in Figure 1 is: segments {160,202}, {280,353} are categorized "still", segments {622, 631}, are categorized "back movement" and segments {783, 790} and {812,819} are categorized "front movement".

3. MOVEMENT OF OBJECT ENSEMBLES

In this chapter we describe the technique that evaluates whether two or more ROIs within a video frame are approaching or moving away along the x, y plane and disparity axis separately. The method is similar to the one described above, but in this case the input signal is the Euclidean distance D between every ROI center and the center of mass of the entire set of ROIs:

$$D(i)=\sum_{i=1}^{k}\sqrt{(\bar{x}_s-\bar{x}_i)^2+(\bar{y}_s-\bar{y}_i)^2}$$, where k is the total number of ROIs in a video frame, ($\bar{x}_i$, $\bar{y}_i$) are the coordinates of the center of the i_{th} ROI and ($\bar{x}_s$, $\bar{y}_s$) the coordinates of the center of mass of all ROIs which is computed as:

$$\bar{x}_s=\frac{\sum_{i=1}^{k}\bar{x}_i}{k} \qquad \bar{y}_s=\frac{\sum_{i=1}^{k}\bar{y}_i}{k},$$

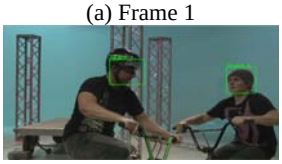
For the depth axis we use the formula below:

$$D_{disp}(i)=\sum_{i=1}^{k}\left|\bar{d}_i-\bar{d}_s\right|,$$

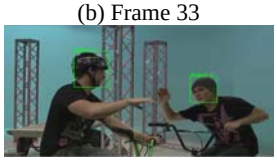
where $\bar{d}_i$ is the mean disparity of the i_{th} object and $\bar{d}_s$ is the mean disparity of all the objects inside the frame. The signals $D(i)$ and $D_{disp}(i)$ are then filtered and approximated by line segments.

(a) Frame 1

(b) Frame 33



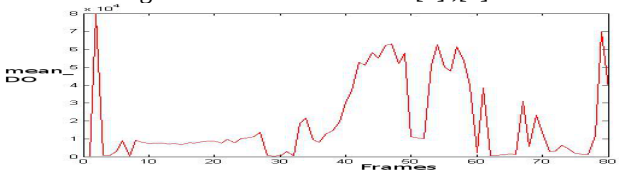
(c) Frame 55



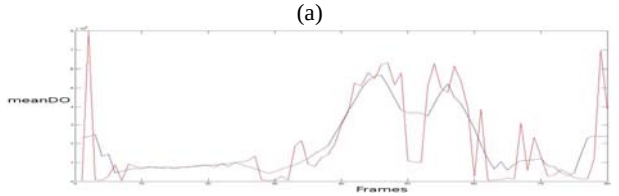
(d) Frame 80

Figure 8. Sample video frames. The green marked ROI rectangles are the output of a face detector and a tracking algorithm.

In Figure 9 and Figure 10 we can see the raw, filtered and approximated signal, for the algorithm that characterizes whether objects approach or move away in the image plane and depth axis respectively, for a video, frames of which are shown in Figure 8. Disparity maps for this video were extracted using the method described in [3] ,[4] .

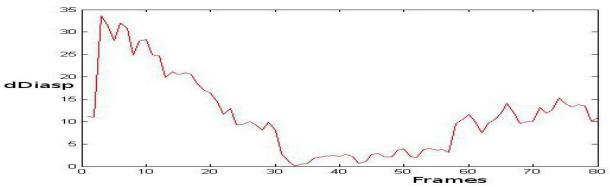


(a)

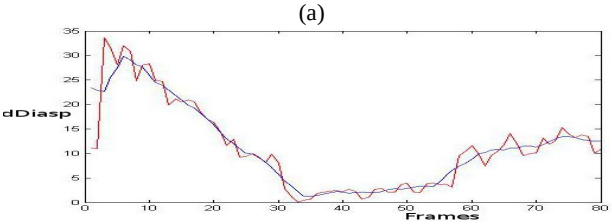


(b)

Figure 9. (a) The signal used as input for the characterization of the image plane movement, (b) the filtered signal and (c) the result of the linear approximation.



(a)



(b)

Figure 10. (a) The signal used as input for the characterization of the depth axis movement, (b) the filtered signal and (c) the result of the linear approximation.

The results of the algorithm are semantically described by a list of $[t_l, s_l, a_l]^T$, $l = 1,2,...,n$, where t_l, s_l, a_l are the start video frame, the duration and the slope of a line segment and n is the total number of segments. Labels "move away xy", "approach xy", "move away depth", "approach depth" and "still" are then assigned to the each segment according to the slope. For the video shown in Figure 8 segments {1, 30},{49, 57}, {62,74} are categorized as "still", segments {30,43}, {57,62}, {74,80} are categorized as "move away xy" and segments {43,49} are categorized "approach xy". Regarding the depth axis segment {1,8} is "move away depth", segment {8,33} is "approach depth" and segment {33,113} is "still".

4. CONCLUSIONS

In this work algorithms that perform semantic labeling of the horizontal, vertical and along the depth axis movement of object/human ROIs have been described. Some representative examples are presented that prove the effectiveness of the algorithms.

5. ACKNOWLEDGEMENT

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement number 287674 (3DTVS). This publication reflects only the author's views. The European Union is not liable for any use that may be made of the information contained therein.

6. REFERENCES

[1] I. Pitas, "Digital image processing algorithms", Prentice Hall international series in acoustics, speech, and signal processing, 1993.

[2] I. Pitas, and A.N. Venetsanopulos, Nonlinear Digital Filters, Boston: Kluwer, 1990.

[3] N. Atzpadin, P. Kauff, and O. Schreer, "Stereo analysis by hybrid recursive matching for real-time immersive video conferencing," IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 3, pp. 321–334, March 2004.

[4] C. Riechert, F. Zilly, and P. Kauff, "Real time depth estimation using line recursive matching," in European Conference on Visual Media Production, November 2011.

[5] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in Proceedings of IEEE Conference of Computer Vision, vol. 1, 2001, pp. 508–515.

[6] G. Bradski, A. Kaehler, and V. Pisarevsky, "Learning-based computer vision with intels open source computer vision library," Intel Technology Journal, vol. 9, no. 2, pp. 119–130, 2005.

[7] O. Zoidi, A. Tefas, and I. Pitas, "Visual object tracking based on local steering kernels and color histograms," IEEE Transactions on Circuits and Systems for Video Technology, 2012.