DEPTH ANALYSIS FOR SURVEILLANCE VIDEOS IN THE H.264 COMPRESSED DOMAIN

H. Nicolas
LaBRI, Univ. Bordeaux, 351 cours de la Libération, 33405 Talence Cedex, France
nicolas@labri.fr

ABSTRACT

Knowledge about the distance of moving objects can be used to enhance the performance of object detection, tracking and classification schemes. However, such information is usually not known a priori. We present an unsupervised method to approximate basic scene geometry properties such as the camera pose in single-view video sequences. At present, the method is working in constraint environments such as traffic surveillance. The proposed approach is solely based on the motion information present in H.264 encoded, compressed video streams and does not rely on object tracking results. We start by constructing motion maps from compressed domain motion vectors. These maps are used to estimate the orientation angle of the camera, which allows to add a depth measure in the form of equidistant lines to the image plane.

Index Terms— *H.264, Compressed domain, Camera pose estimation, Scene geometry*

1. INTRODUCTION

Numerous computer vision tasks like object detection, tracking, scene segmentation and behavior analysis can benefit from information about the scene and its basic geometry. Perspective projection obscures the relationships that are present in the actual scene. Perspective plays an important role not only because it affects the size of the object's projection on the image plane, but also for the estimation of its speed for example. Approximate knowledge of the 3D scene geometry can provide very useful information during analysis tasks.

Another aspect that plays an increasingly important role in computer vision is the consideration of context. Although sophisticated object detectors, classifiers and scene segmentation schemes exist, such tasks still remain challenging research problems. Information about the context can deliver useful information to enhance detection results. The presented approach is inspired by Hoiem et al. [1], in which the authors identify three crucial elements that are required for scene understanding: (i) object detectors, (ii) approximate camera position and orientation, (iii) rough 3D scene geometry. The estimation of geometrical properties like the position of the road, the orientation of the camera pose or the distance of moving objects can be approached in a variety of different ways. The largest family of methods is based on multiview sequences or stereo vision, such as [2, 3]. Except from specialized applications like robot vision or multi-view video surveillance, the majority of real-world applications employs single-camera setups. A family of algorithms on monocular sequences is grouped under the keyword Structure-from- Motion (SfM), where ego-motion and changes in perspective are used to infer the 3D structure of the scene or of moving objects, e.g., [4, 5].

Previous automatic camera rectification approaches on monocular sequences are based on rich information provided on pixel level. To name a few, Bose and Grimson [6] proposed a rectification scheme by observing objects which move at constant velocity for some part of their trajectory. Lin et al. [7] determined the camera orientation through the vanishing point, which is obtained by analyzing the bounding box sizes of detected persons. A survey of image domain self-calibration techniques is presented in [8]. Analysis on pixel level enables accurate camera pose estimates, but approximate results can deliver sufficient information for certain tasks like object classification [1]. Only very few compressed domain attempts have been published on this topic. Mbonye [9] uses MPEG-2 compressed domain data to adjust the camera pose in road traffic application with car mounted cameras.

In [10] we presented an approach to estimate the camera orientation based on compressed domain tracking results of moving objects. Since the estimation itself relies on the results of a processing pipeline with multiple stages, many error sources are introduced, which leads to very rough approximations of the camera orientation. In this article, we propose a more robust compressed domain method for constraint environments such as traffic surveillance. An overview of the proposed scheme is shown in Fig. 1. The different elements of the processing chain are described in the following sections.

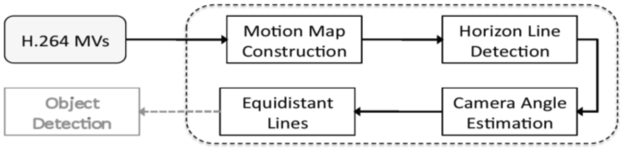

Fig. 1 : Overview of the proposed system

2. ROAD AREA AND DIRECTION ESTIMATION

Traffic videosurveillance systems often use an MPEG-type video streams such as H.264. During encoding, each picture is sub-divided into so-called macroblocks (MBs). Most MBs in P- and B-frames are predicted from past or future reference frames. Motion vectors (MVs) pointing to the position of the current MB in the respective reference picture are present in the bit stream. These MVs can be regarded as a sparse and noisy version of true scene motion. MVs can be easily extracted from the bit stream, only the entropy coding has to be reversed and full stream decoding can be avoided. The idea is therefore to reuse these vectors in order to reduce as much as possible the computational load. Using the extracted MVs as input, we construct a gray-scale motion density map of the scene by counting the number of non-zero MVs at all MB and pixel positions. At each pixel *p*, the motion intensity map (MAP) is constructed by incrementing the value (MAP(p)) with each incoming non-zero MV. MAP is normalized by the number of used input frames. The main direction lines *(MDL)* are therefore estimated by maximizing the following energy function:

$$MDL(L) = argmax \sum_{p\in L} \quad MAP(p)$$

with *{L}* the set of lines crossing the image. If *N* significant maximum of the MDL function are detected, it means that *N* different traffic directions are detected. Fig. 2 shows examples of estimated motion maps and detected directions lines. If the intersection of two of these MDL are located on or above the estimated vanishing line (see section 3), it means that they represent the same (or parallel) road (see images a, b, and d on Fig. 3). In this case, a line MDLA representing the average direction of two MDL is created. If it is not the case it means that they represent two different roads (see Fig. 3.c). The construction time for the motion maps has to be sufficiently long so that moving objects occurred throughout the scene. The average training time is 30 seconds in normal daytime and traffic conditions.

A MAP represents a rough approximation of the frequented ground surface. We obtain a single, binary mask *R* of the road by thresholding the motion map. Holes in the resulting binary mask are filled through morphological filtering. A last post-processing step consists of rejecting all blobs that are situated entirely above the estimated vanishing line. Road segmentation results are provided in Fig. 6. All important parts of the road have been captured; only the security lane in sequence 2 is cut off because no vehicle used it during the training period.

3. VANISHING LINE ESTIMATION

The vehicles vanishing lines (*VL*) correspond to the distance up to which moving vehicles can be sensed. The *VL* delivers important information about the orientation of the camera with respect to the ground plane, i.e., the road in our case. A vanishing line can theoretically be outside the image. *VL* can be efficiently approximated by analysing the constructed motion maps. To achieve this, the descriptor *V* is defined as:

$$V(I) = \sum_{p\in(VL(I)\cap R)} \quad MAP(p)$$

(1)

where *VL(I)* is a segment centered around and perpendicular to the corresponding MDL and MDLA*,* and crossing it at point *I*. Theoretically, a *VL* is estimated at the position corresponding to *VL(I)=0* (no more detected vehicle's displacements). In practice, the used H264 motion fields are noisy. As a consequence, we consider that a *VL* corresponds to the value for which *L(i)* drops under 3% of its maximal value. This threshold has been empirically fixed. Its value is not sensitive and very similar results are obtained for a range around of 1 to 5%. See examples of the energy function *VL(I)* and the estimated *VL* in Fig. 3. The method delivers good approximations for all test sequences even for the rather complex ones.

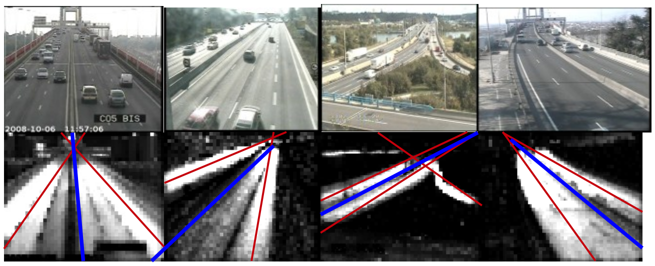

Fig. 2. Examples of screenshots (in different weather conditions (cloudy (a), rainy (b) and sunny (c,d))) and motion density maps. Red lines represent the estimated direction of the circulation lanes (MDL). The blue lines represent the MDLA.

4. CAMERA POSE ESTIMATION

In this section, we show how the camera orientation can be estimated given the position of the vanishing line. Figure 5 shows the assumed scene setup. If no lens distortion is present, the relationship between the object's distance z to the camera (modelled as pinhole) and the vertical bottom position on the image sensor *ys* can be expressed as

$$y_S = \frac{d}{2} + f * \tan(\tan^{-1}(\frac{z}{h_{cam}}) - \alpha_{cam}), \qquad (2)$$

where *d* is the vertical dimension/height ratio of the image sensor, *f* is the focal length, h_{cam} denotes the camera height and α_{cam} the orientation angle of the camera, relative to the ground plane. An α_{cam} of 0° corresponds to bird's eye view and 90° means the camera is parallel to the ground. Under the flat ground assumption, the position of the vanishing line on the image sensor is given at $z \rightarrow \infty$. Since

$$\lim_{z\to\infty} \tan^{-1}(\frac{z}{h_{cam}}) = \frac{\pi}{2} = 90° \qquad (3)$$

the vertical position of the vanishing line VL_Y is given as

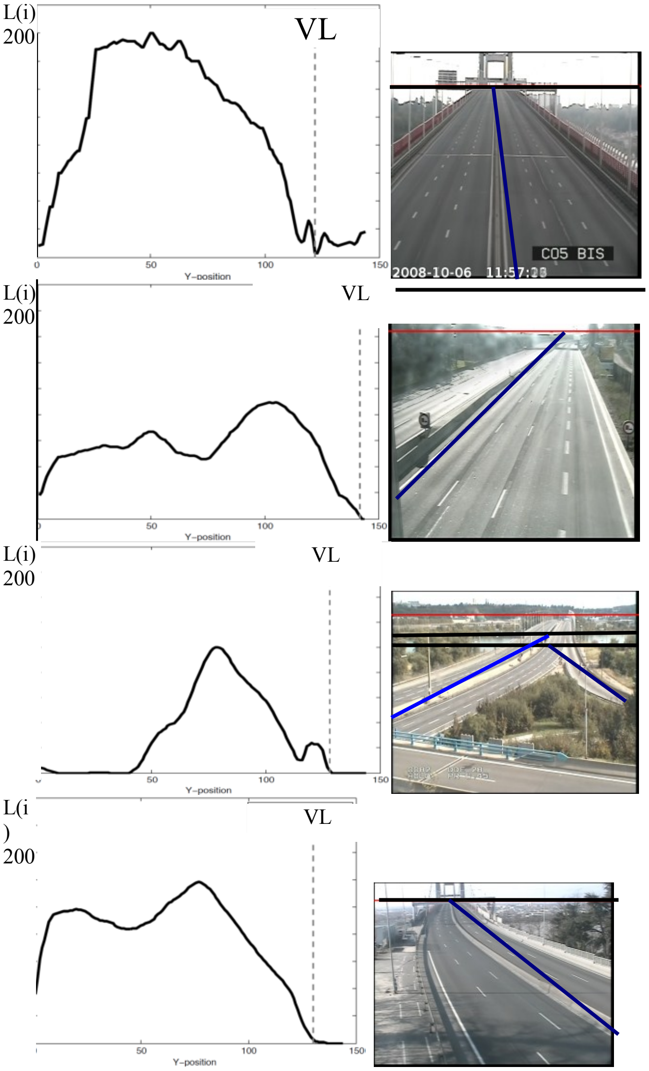

Fig. 3. Left: I vs L(i). Right: estimated positions of the VL (in black). Red lines representing the real VL (ground truth) are displayed when different from the estimations. In the third sequence, two VL lanes have been detected .

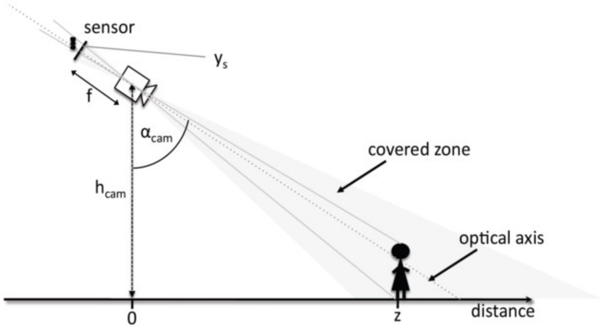

Fig. 5. Camera setup

$$VL_Y = \frac{d}{2} + f * \tan(\frac{\pi}{2} - \alpha_{cam}) \qquad (4)$$

what approves that the position of the vanishing line in the image plane depends only on the orientation angle of the camera, the dimension of the image sensor and the focal length, but not on the camera height [11]. For cameras that are parallel to the ground (α_{cam} = 90°), potential lens distortions and the focal length also loose their influence, because the horizon always appears at d/2, i.e., in the center of the image. Equation 4 can be rewritten as

$$\alpha_{cam} = \frac{\pi}{2} - \tan^{-1}(\frac{VL_Y - d/2}{f}) \qquad (5)$$

so α_{cam} can be determined if the vertical position of the vanishing line VLy and the focal length f are known. It should be pointed out that this last assumption is realistic in the specific case of traffic videosurveillance applications for which the focal *f* can be known when the camera are installed.

5. DISTANCE INDICATION

Estimates of the vanishing line and the camera orientation allow us to project imaginary equidistant lines from the ground plane onto the image plane, hence adding depth information to the sequence. The green lines in Fig. 6 correspond to equidistant lines on the ground plane. They have been obtained through Eq. 2 at the estimated camera angle where the distance mapping function for the given camera angle is sampled at evenly spaced positions. The parameter camera height can be set arbitrarily and does not influence the calculation, since it only rescales the x-axis but does not change the curvature of the function. Equidistant lines add a depth measure to the 2D image plane that can be used to assist object tracking, detection and classification algorithms. If lane markers are visible, the lines can also be used to evaluate the quality of the camera angle estimation (if no ground truth is available).

Our test sequences have been acquired with cameras used in a traffic videosurveillance system with d = 24 mm and f=40 mm. Table 1 gives results for four test sequences. The ground truth has been obtained using the manual calibration technique proposed by Worrall et al. in [12], which is based on parallel lines on the ground plane. The proposed method bypasses any potentially erroneous object detection and tracking steps, hence the results are more accurate and stable than based on moving objects. A visual verification of the results can be carried out by drawing equidistant lines in the image plane, as illustrated in Fig. 6. Each segment should ideally contain the same number of lane separation markers, which is approximately the case. If the focal length of the camera is unknown, the error which is introduced from fixing it at a standard value is limited. In our example, if f is varied by ±5mm, α_{cam} changes by only ±1.8°.

6. CONCLUSION

For the specific use-case of highway surveillance, we presented a simple and effective method to approximate the vertical position of the vanishing line and to segment the road portion in the image plane. Based on the position of the vanishing line, a robust approach to estimating the camera orientation was provided. The method has been tested on 35 video sequences acquired at different highway locations. In 94% of the cases, the vanishing line and the estimated camera orientation is accurate enough to add depth information to the scene, which is crucial regarding applications like the classification of vehicles according to their size, or the perspective correction of speed measurements.

REFERENCES

[1] D. Hoiem, A. Efros, and M. Hebert, “Putting objects in perspective,” Int. Journal of Computer Vision (IJCV), vol. 80, no. 1, pp. 3–15, 2008.

[2] O. Faugeras, Q.-T. Luong, and T. Papadopoulou, The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications, MIT Press, Cambridge, MA, USA, 2001.

[3] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge Univ. Press, March 2004.

[4] S. Soatto and P. Perona, “Reducing ”Structure From Motion”: a general framework for dynamic vision part 1: Modeling,” IEEE Trans. on Pattern Analysis, vol. 20, no. 9, pp. 933–942, 1998.

[5] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, An Invitation to 3D Vision: From Images to Geometric Models, Number 26 in Interdisciplinary Applied Mathematics Series. Springer Verlag, New York, LLC, 2003.

[6] B. Bose and E. Grimson, “Ground plane rectification by tracking moving objects,” in IEEE Int. Workshop on Visual Surveillance and PETS, 2004.

[7] S.F. Lin, J.Y. Chen, and H.X. Chao, “Estimation of number of people in crowded scenes using perspective transformation,” IEEE Trans. on Systems, Man, and Cybernatics, vol. 31, no. 6, pp. 645 -654, November 2001.

[8] E. Hemayed, “A survey of camera self-calibration,” IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS’03), p. 351, 2003.

[9] K.P. Mbonye and F.P. Ferrie, “Attentive visual servoing in the MPEG compressed domain for uncalibrated motion parameter estimation of road traffic,” in Proc. of Int. Conf. on Pattern Recognition (ICPR’06), Washington, DC, USA, 2006, pp. 908–911, IEEE Computer Society.

[10] C. K¨as and H. Nicolas, “Rough compressed domain camera pose estimation through object motion,” in IEEE Int. Conf. on Image Processing (ICIP 2009), Cairo, Egypt, November 2009.

[11] Peter Ward, Picture Composition, Focal Press, 2002.

[12] A. D. Worrall, G. D. Sullivan, and K. D. Baker, “A simple, intuitive camera calibration tool for natural images,” in Proc. Of the Conf. on British Machine Vision (BMVC’94), Surrey, UK, UK, 1994, vol. 2, pp. 781–790, BMVA Press.

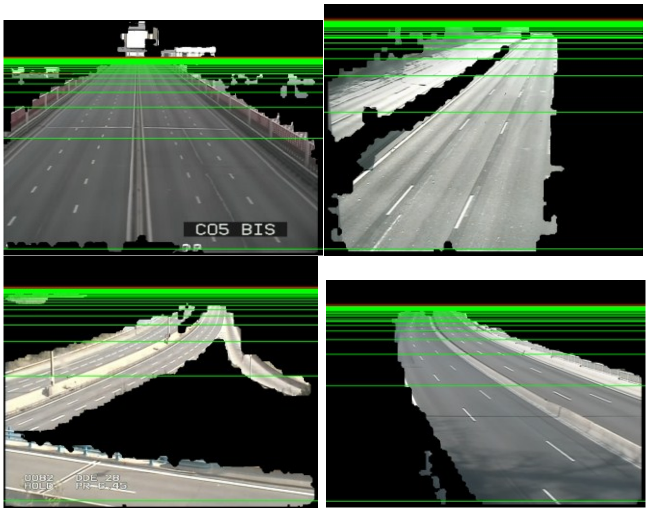

Fig. 6. Compressed domain road segmentation results with equidistant lines in green

	Seq 1	Seq 2	Seq 3	Seq 4
α_{cam} [proposed]	79.5°	75.8°	75.6 °	77.6 °
α_{cam} Ground truth [12]	78.7°	77.5 °	77.4 °	76.2 °

Table 1. Camera orientation results