IMPROVED 2D-TO-3D VIDEO CONVERSION BY FUSING OPTICAL FLOW ANALYSIS AND SCENE DEPTH LEARNING

José L. Herrera *Carlos R. del-Blanco* *Narciso García*

ABSTRACT

Automatic 2D-to-3D conversion aims to reduce the existing gap between the scarce 3D content and the incremental amount of displays that can reproduce this 3D content. Here, we present an automatic 2D-to-3D conversion algorithm that extends the functionality of the most of the existing machine learning based conversion approaches to deal with moving objects in the scene, and not only with static backgrounds. Under the assumption that images with a high similarity in color have likely a similar 3D structure, the depth of a query video sequence is inferred from a color + depth training database. First, a depth estimation for the background of each image of the query video is computed adaptively by combining the depths of the most similar images to the query ones. Then, the use of optical flow enhances the depth estimation of the different moving objects in the foreground. Promising results have been obtained in a public and widely used database.

Index Terms — 2D-to-3D conversion, depth maps, depth prior, clustering, machine learning

1. INTRODUCTION

In the last decade, an important increment in the number of 3D players and displays such as smartphones, TVs, cinemas, DVD/Blu-Ray or video game consoles has been experimented. However, the amount of 3D content, such as images, videos or TV broadcasting, has not increased at the same rate, thus creating an important gap between the volume of available 3D content and the quantity of 3D players. To compensate this situation, different algorithms to automatically or semi-automatically convert the current 2D content into 3D have appeared to fulfill the demand of the users for 3D experience.

The 2D-to-3D image and video conversion task is a procedure that is usually divided in two main stages. In the first one, a depth map estimation of the image or video is computed, and then, with this estimation and the original image a stereoscopic pair is built. In this paper, we focus in the first stage of the task: the depth extraction from a single color image, which is more challenging, and moreover, there currently exist many algorithms for generating a stereo-pair achieving good quality.

In last years, new algorithms based on machine learning principles have appeared as an alternative for the 2D to 3D image and video conversion process. The main hypothesis behind these new family of methods is that those images which have a high structural similarity in color will probably have a similar depth structure. Saxena et al [1] developed a supervised learning strategy to infer the structure of the scene from a single image using an image parsing strategy and Markov Random Fields to determine 3D locations and orientations. The incorporation of semantic labels and more sophisticated methods in [2][3], improved scene depth results. An approach based on Scale Invariant Feature Transform (SIFT) flow and an optimization post-process was used by Karsch et al.[4] to improve the results and extend the method to work with videos. Using a descriptor based on Histogram of Oriented Gradients (HOG) to find similar images instead of SIFT flow, and a Cross Bilateral Filter to enhance the resulting depth map, Konrad et al. [5] presented a more computationally efficient method. We used in a previous work [6] a new approach to find an adaptive number of similar images and to fuse them in a weighted way to infer the depth of the scene. As the computational cost of this methods is proportional to the database size these algorithms are impractical when using large datasets. To alleviate this situation, we proposed in [7] an algorithm that performed a hierarchical search in a previously clustered database., improving significantly the efficiency of the search process. In [8] we used GIST as a descriptor since it improves the performance of the algorithm, and introduced edge-based refinement post-processing to enhance the 3D structure of the scene.

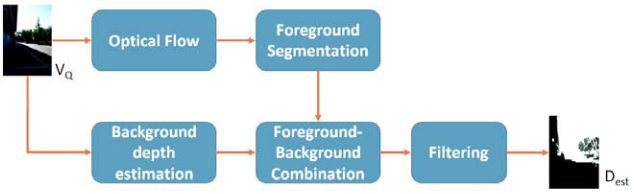

Figure 1. Overall block diagram of the proposed 2D-to-3D image and video conversion algorithm

While all the previously mentioned methods try to solve the 2D-to-3D conversion for images, the video conversion task has not been so widely discussed by the machine learning based algorithms due to some other strategies such as Structure from Motion (SfM) achieve very good results in the case of video sequences. In this paper, we extend the learning based approach for images to video in those sequences with a static camera that are not so well solved by techniques such as SfM. The algorithm is divided in three main parts. In the first one, a depth estimation of the background is computed by combination of depth maps of similar images found in a clustered color + depth database trough a GIST-based descriptor. In the second part, the foreground is segmented analyzing the optical flow to manage the different objects individually. In the last step of the algorithm, the depth estimation of the background is combined with the foreground information to obtain a depth prior of each frame of the video sequence, and then, this depth prior is filtered using a Cross Bilateral Filtering to enhance the edges of the final depth estimation.

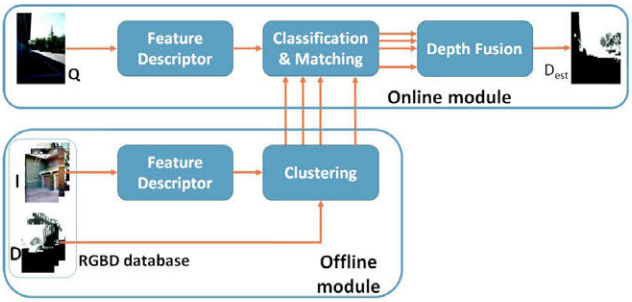

Figure 2. Detail diagram of the Background depth estimation module

2. ALGORITHM DESCRIPTION

Given a query sequence V_Q, and a database DB composed by pairs of color images I and their associated depth maps D, the aim of the presented approach is the extraction of depth maps D_{est} of all the frames in V_Q. The correct performance of the algorithm is subject to the presence of structurally similar images to the query frames Q in DB.

The proposed 2D-to-3D video conversion algorithm, can be divided into four main stages and its overall block diagram can be shown in Fig. 1. In the first stage, a clustering of all color images in DB is performed to make clusters containing photometrically similar images. In the second stage, each frame of the query video sequence is classified into one of this clusters, and matched with all the images that compose the selected cluster to find the most similar images in DB to the query frame Q. The depth maps associated to these similar images are fused to build a depth estimation of the scene background D_{bg}. In Fig. 2 the detailed scheme of this stage is shown. In parallel, in the third stage an optical flow analysis of the scene is used to detect and extract the moving objects in the foreground of the scene. Finally, the background and the foreground are combined to compute a depth prior of the scene D_{prior}. Then, a Cross Bilateral Filtering is applied to enhance the edges and obtain the final depth estimation D_{est}.

The main differences with our previous approaches [6][7] [8] are the extension of the method from only images to video sequences using optical flow to detect moving objects in the scene, the independent computation of background and foreground instead of obtaining the global depth in one step, and the computation of just one background for the whole video sequence instead of doing it for every frame as the previous methods do, thus improving the computational efficiency of the algorithm.

2.1. Feature descriptor

Color images in the database DB with similar structure to the query image Q are used in the depth estimation process. To find out which images in the dataset are similar to the query image, we characterize the images by a feature descriptor that represents the structure of the image. This image feature descriptor is based on GIST [9], which provides a compact representation of the image structure. The overall descriptor is computed by dividing the image into 16 tiles (4 horizontally and 4 vertically), and obtaining a GIST descriptor per tile. Then, for image I, the descriptors of every tile are stacked in a single vector $F(I)$, which characterizes the whole image:

$$F(I) = [\overline{GIST(t_1)} \quad \overline{GIST(t_2)} \quad ... \quad \overline{GIST(t_{16})}], \qquad (1)$$

where $\overline{GIST(t_i)}$ is the descriptor of the tile i of the image

For the whole dataset DB, these descriptors are pre-calculated off-line before the beginning of the conversion process. In the case of each query frame Q, the feature descriptor is computed as indicated for every frame $F_Q(n)$ of a sequence, and then, the median of the resulting descriptors is obtained to minimize the effect of the moving objects in the sequence, building the feature descriptor corresponding to the background of the video sequence F_{bg}. Thus, we avoid to compute one background depth per image as the other state-of-the-art approaches do, resulting in a significant improvement in computational efficiency. This task is computed online at the beginning of the process.

$$F_{bg} = median([F_Q(1), F_Q(2), ..., F_Q(N_{seq})]), \qquad (2)$$

2.2. Database Clustering

The database clustering process has a double motivation. First, the organization of all the images in the dataset by structural similarity. In parallel, as described in [7], the division of the database into clusters allows the algorithm to work more efficiently for large datasets. The clustering process is performed over the feature-based representation of the color images in the dataset computed in section Section 2.1. This part of the algorithm can be performed offline, before a query video sequence V_Q arrives for conversion.

In order to cluster the images according to their feature-based representation in such a way that images grouped in the same cluster have a similar structure, we employ the K-means algorithm along with the correlation coefficient, as similarity metric. The centroid of each cluster F_C is the average across all the feature descriptors of the color images in the cluster

2.3. Classification and Matching

The GIST descriptor of the background of the query video sequence F_{bg} is compared with the centroids of the clusters F_C in which the database DB is divided to classify the sequence into one cluster. The metric used for this classification is the correlation coefficient between the descriptors:

$$c(n) = corr(F_{bg}, F_{Cn}), \qquad (3)$$

where $corr()$ is the correlation measure, F_{bg} is the GIST-based image feature descriptor of the background Q, and F_{Cn} is the n^{th} centroid of the clustered database DB.

Once F_{bg} has been classified, this descriptor is matched with all the descriptors of the images that belong to the selected clusters and the depth maps associated to the k images with a higher similarity are selected to be fused. The metric used for measuring the similarity is again the correlation coefficient in the same way as was exposed in Eq. 3

2.4. Depth fusion

The selected depth maps in the previous stage are combined to obtain the estimation of the depth of the background of the scene D_{bg}. The more similar an image is to the query Q, the higher should be its depth contribution to the background depth estimate. Specifically, each depth map is weighted by the correlation coefficient computed in the previous stage as follows

$$D_{bg} = \sum_{n=1}^{k} c(n) D_n, \qquad (4)$$

where D_{bg} is the depth map of the sequence background, D_n is the depth map associated with image I_n and k is the number of images selected by similarity search in the previous stage. The resulting D_c is a preliminary depth estimate of Q.

2.5. Foreground inclusion

To extract the foreground from a query video sequence V_Q we have computed the optical flow of the sequence as described in [10]. Then a threshold is applied in such a way that only pixels with an absolute value of optical flow greater than this value are considered to be moving pixels and a binary mask is created with the pixels classifieds as in movement.

Now, a depth prior D_{prior} is built with the information of the foreground objects, and the previously obtained depth estimation of the background D_{bg} as follows.

$$D_{prior}(x,y) = \begin{cases} D_{bg}(x,y) & \text{if } (x,y) \in R_s \\ \alpha D_{bg}(x,y) & \text{if } (x,y) \notin R_s \end{cases}, \quad (5)$$

where R is the region that groups all the pixels selected as foreground.

2.6. Edge Enhancement

After the combination of the foreground and the background, a globally consistent depth estimation is obtained. However, the result is too smooth and presents local inconsistencies around the edges due to the fusion of the k most similar images. In order to maintain the global result while enhancing the edges, and aligning them respect to the original edges of the query frames Q, a Cross Bilateral Filtering is applied.

Cross Bilateral Filtering is a variant of bilateral filtering where the Gaussian function is controlled by an external intensity image. In this case, the query frame Q is used to control the smoothing. Moreover, Cross Bilateral Filtering reduces the noise in homogeneous areas, and align and enhance the edges of the estimated depth prior regarding to Q. Formally, it can be expressed as:

$$\begin{aligned} W(x) &= \sum_y h_d(x-y) h_Q(Y(x) - Y(y)) \\ D_{est} &= \frac{1}{W(x)} \sum_y D_c(y) h_d(x-y) h_Q(Y(x) - Y(y)), \end{aligned} \quad (6)$$

where D_{est} is the final estimated depth map of a query frame Q, $h_d(x)$ and $h_Q(x)$ are Gaussian functions, and $Y(x)$ is the intensity value of pixel x in image Y. The Gaussian function over the position $h_d(x)$ is calculated over the depth map image, while the Gaussian function over the intensity $h_Q(x)$ is computed over the query image Q. The depth map is smoothed as a result of this filtering, but preserving and enhancing the edges of the query frame.

3. EXPERIMENTAL RESULTS

The proposed approach has been tested using a subset of the StereoRGBD Indoor1 database [4]. This subset consists in all the video sequences of the whole database where the camera is static and it is composed by 6967 color images + their associated depth maps.

The resolution of the color images and depth maps is 579×430. Nevertheless, like Konrad [5], color and depth images have been resized to a 320×240 resolution for computational efficiency and for a straightforward comparison with with other approaches in the state of the art.

The quality of the final estimation is sensible to the value of the number of clusters in which the databases are divided n_c and to the number k of depth maps fused to build the background depth estimation D_{bg} in Section 2.4. We have chosen a value of $n_c = 80$ and $k = 15$ since these values result in both cases in a good trade-off between good performance and time efficiency.

Algorithm - metric	RMSE	PSNR	C
HOG Learning Based [5] (2012)	0.26	11.9	0.43
Adaptive LBP-based [6] (2014)	0.24	12.6	0.55
Hierarchical Search [7] (2014)	0.25	12.6	0.51
Background	0.24	12.5	0.57
Background + foreground	**0.23**	**13.2**	**0.60**

Table 1. Evaluation of state-of-the-art algorithms using the RMSE, PSNR and Correlation Coefficient (C) metrics in the static camera sequences of the StereoRGBD Indoor1 database in Leave One Out configuration. The results are the average over the 6967 test images.

To evaluate the performance of the algorithm quantitatively, we performed the tests in a leave-one-out cross-validation configuration removing from the train database all the images belonging to the same sequence that the query image belongs. As the quality metric, we used the Root Mean Square Error (RMSE), the Peak Signal to Noise Ratio (PSNR), and the correlation coefficient (C), all of them computed between the depth ground truth and the depth estimation provided by this method.

The RMSE and the PSNR are mathematically expressed by

$$\begin{aligned} RMSE &= \sqrt{\sum_i \left(D_Q(i) - D_{est}(i)\right)^2 / N}, \\ PSNR &= 20 \log_{10} \frac{max(D_Q)}{RMSE}, \end{aligned} \quad (7)$$

where N is the number of pixels in the query image Q, D_Q is the ground-truth depth of Q, D_{est} is the final depth estimation, and max is a function that return the maximum value. These metrics are depending on the range of values that the images have. In this case, the depth range has been set to $[0, 1]$

The correlation coefficient is defined as follows:

$$C = \frac{\sum_i (D_{est}[i] - \mu_{D_{est}})(D_Q[i] - \mu_{D_Q})}{N \sigma_{D_{est}} \sigma_{D_Q}}, \quad (8)$$

where N is the number of pixels in $\mu_{D_{est}}$ and D_Q (ground-truth depth of the query image Q), $\mu_{D_{est}}$ and μ_{D_Q} are the empirical mean values of D_{est} and D_Q, respectively, $\sigma_{D_{est}}$ and σ_{d_Q} are the corresponding empirical standard deviations, and it refers to each pixel of the image. The normalized cross-covariance C takes values from -1 to +1 (values close to +1 indicate that the depth maps are very similar an values close to -1 suggest they are complementary).

The results of our approach have been compared with the HOG-Based Depth Learning approach of Konrad et al [5], and with our previous works: adaptive LBP-based approach [6] and the hierarchical search approach [7]. Table 1 shows the quantitative results for StereoRGBD Indoor1 [4] database. For the proposed approach, we show the results of the generated depth estimation for just the background estimation, (without temporal information) and after adding the information provided by the optical flow analysis to better observe the effect of introducing the temporal information in the algorithm.

As can be seen, we outperform the other compared algorithms before the introduction of the temporal information in our estimation for the three used metrics (RMSE, PSNR and C). These results become even better for each metric after combining the information for both background and foreground.

In Fig. 3, some examples of query frames, the optical flow binary mask, background depth map, final depth map estimation

Figure 3. From left to right: Query Frame, Optical Flow binary mask, Background Depth Estimation, Final Depth Map Estimation and Depth Ground Truth for images of StereoRGBD Indoor1 database

and depth groud truth, generated by this method, are shown.

4. CONCLUSIONS

A novel algorithm for automatically estimate the 3D structure of a query video sequence has been presented in this paper. The approach uses a machine learning framework to globally estimate the depth of the background of the scene while an optical flow analysis is used to extract the foreground and manage the moving objects locally. The method uses K-means to divide the database into different clusters to easily find the most similar images in the dataset and also to extend the use of the algorithm to large databases avoiding the problem of excessively long times that present this kind of algorithms for the search stage. The query image is matched with all the cluster centroids and the most similar images in this cluster are combined to build the depth of the background. In this background we insert the objects in the foreground, segmented through optical flow, by highlighting the depth of the region corresponding to the foreground. The algorithm outperforms other state-of-the-art approaches while keeps the computational cost in a low level by using a hierarchical search and also by computing only one background depth estimation for each video sequence instead of one per image as others do.

5. REFERENCES

[1] A. Saxena, H. Chung Sung, and Y. Ng Andrew, "Learning depth from single monocular images," in *NIPS 18*. 2005, MIT Press.

[2] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing: Label transfer via dense scene alignment," in *IEEE Conf. on Comput. Vis. and Pattern Recognit., CVPR 2009.*, June 2009, pp. 1972–1979.

[3] B. Liu, S. Gould, and D. Koller, "Single image depth estimation from predicted semantic labels," in *IEEE Conf. on Comput. Vis. and Pattern Recognit., CVPR 2010.*, June 2010, pp. 1253–1260.

[4] K. Karsch, C. Liu, and S. Kang, "Depth extraction from video using non-parametric sampling," in *Computer Vision ECCV 2012*, 2012, vol. 7576 of *Lecture Notes in Computer Science*, pp. 775–788.

[5] J. Konrad, M. Wang, P. Ishwar, C. Wu, and D. Mukherjee, "Learning-based, automatic 2D-to-3D image and video conversion," *IEEE Trans. on Image Process.*, vol. 22, no. 9, pp. 3485–3496, Sept 2013.

[6] J.L. Herrera, C.R. del Blanco, and N. Garcia, "Learning 3D structure from 2D images using LBP features," in *IEEE International Conference on Image Processing*, October 2014, pp. 2022–2025.

[7] J.L. Herrera, C.R. del Blanco, and N. Garcia, "Fast 2d to 3d conversion using a clustering-based hierarchical search in a machine learning framework," in *IEEE 3DTV-Conference*, July 2014, pp. 1–4.

[8] J. L. Herrera, C. R. del Blanco, and N. Garca, "Edge-based depth gradient refinement for 2d to 3d learned prior conversion," in *2015 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, July 2015.

[9] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[10] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 978–994, May 2011.