ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

Automatic Conversion of Image and Video from 2D to 3D with Steganographic Data Hiding in Converted 3D Image

Sariga N P[1], Sajitha A S[2]

Department of ECE, Nehru College of Engineering and Research Center, Thrissur, India[1,2]

Abstract: We are going to implement steganographic data hiding process in the converted 3D image. Even if a large growth has happened during last some years in this area of image conversion, the 3D content's availability is still remain less by that of the 2D counterpart. We consider the problem of estimating detailed 3D structure from a single still image of an unstructured environment. Here the introducing idea is to create 3D models which are accurate quantitatively and also visually pleasing. To address the existing limitation, many image conversion methods have been proposed. The conversion by global nearest neighbor method is one of the effective way for 2D to 3D conversion. Here this method is used for the conversion process. The results demonstrate that repositories of 3D content can be used for effective 2D to 3D image conversion. We can also hide the data in 3D image by using steganographic data hiding process. So as to gain more security and efficiency than usual 2D image data hiding, Steganography is one of the efficient methods for securely hiding the data . It is a type of secret communication and security system used in hiding secret data inside digital mediums. An extension to video can be implemented by temporal continuity of computed depth maps.

Index Terms: 3D Image, Image Conversion, Stereoscopic Images, Nearest Neighbor Classification Method, Cross Bilateral Filtering, Steganography.

I. INTRODUCTION

The 3D image provides more information than the 2D image. The 3D image gives better real time world experience than 2D image. There are two steps in 2D to 3D conversion process: depth estimation for a given 2D image and depth-based rendering of a new image in order to form a stereo pair. While the rendering step is well understood and algorithms exist that produce good quality images, the main problem is in estimating depth from a single image or video frame. So we are focusing on depth recovery and not on depth-based rendering.

One of the old methods of conversion is based on learning a point transformation that relates local low-level image or video attributes at a pixel to scene-depth at that pixel. Once the point transformation is learned, it is applied to a monocular image, i.e., depth is assigned to a pixel based on its attributes. Here we are using the conversion method of Global Nearest Neighbor Depth Learning method to address this limitation. In this method that estimates the global depth map of a query image or video frame directly from a repository of 3D images using a nearest-neighbor regression type idea [1].

In which method, among millions of 3D images that are available on-line, there exist many whose 3D content exactly matches that of a 2D input (query) we wish to convert to 3D. And we are using this cluster of similarly matched 3D images as the repository of images used for further process on the query image. We have an assumption that two images which are photo metrically similar also have similar 3D structure (depth). This is not unreasonable since photometric properties are often correlated with 3D content (depth, disparity).

We are implementing Steganographic data hiding method in this converted 3D image for getting more security and efficiency from the traditional 2D image data hiding methods.

Steganography is a process of data hiding in efficient and secured data communication. It can be described as the art of concealed communication, which means, hide and send the data through apparently harmless carriers in order to conceal the presence of secret information. Steganography is the word derived from the Greek word “steganos” that means “covered” and “graphei” which indicates “writing”. Thus the two words get combined to form steganography, which means “covered writing.

II. AUTOMATED METHOD

Main step in 2D to 3D conversion process is the estimation of depth of the query image. There are many ways to estimate the depth of a 2D query image. Estimating the shape from the shading problem is one of the examples for that. For only some special cases, the estimates of quality depth can be found out. In other methods like frequently called multi-view stereo, attempt to recover depth by estimating scene geometry from multiple images not taken simultaneously. For example, a moving camera permits structure-from-motion estimation while a fixed camera with varying focal length permits depth-from-defocus estimation. These both are examples of the use of multiple images of the same scene which are captured at different times and also under different exposure conditions. For example, all images of the Statue of Liberty. Even if such methods are similar with the method proposed, there exist some differences between them. The major difference is that we are using all images

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

available in a large repository in the case when we use images for these methods known to depict the same scene as the query image [12, 15] and these are automatically select suitable ones for depth recovery.

The 2D to 3D video conversion (2D to 3D stereo conversion) is the procedure of transforming the 2D flat film in to 3D film format, which in most cases is stereo format. So we can define stereo conversion is that the process of imagery creation for each of our eyes from one 2D image. The 2D to 3D conversion gives the binocular disparity depth cue to digital images which are perceived by our human brain. So, if we done it in a correct way along with satisfying all its required condition, it makes better the immersive effect when viewing the stereo video in comparison to the 2D video. For the two films, which under gone revisiting the original computer data and it took almost four months, and also took an additional six months in order to add the 3D.

Several electronics manufacturers have developed 2D-to-3D converters which perform in real time. The working basics of all those designs rely on stronger assumptions and simpler processing, which are much simpler than the methods which are discussed above. For example, the objects closer to the viewer looking like they are moving faster or very larger and also the objects located further away is assumed to have higher frequency of texture. It is work very well in specific scenarios, even if it is very difficult to make heuristic assumptions that cover all possible background. Even if it is not impossible, there will have some difficulties in its way of process.

In the method that fuses SIFT-aligned depth maps which selected from a large 3D database, even if this approach proved to be computationally demanding [9]. Here we neglected depth alignment which is on the SIFT-based because it is very costly and used a different metric for selecting most similar depth fields from a database. We observed no significant quality degradation but a significant reduction of the computational complexity [10]. In the paper proposed by Karsch [8], the depth extraction method based on SIFT warping have been introduced. It follows an approach which has unnecessarily initial complexity to depth extraction [9].

III. PROPOSED METHOD

A. Conversion from 2D To 3D By Global Nearest Neighbor Depth Learning

The 2D to 3D conversion based on learning a local point transformation has the undisputed advantage of computational efficiency the point transformation can be learned off-line. It applied basically in real time the same transformation is applied to images with potentially different global structures of the 3D scene structure. This type of conversion is based on purely local image or video attributes. This image or video attributes are color, motion and spatial position at each of the pixel.

To address these limitations, we have a method that estimates the global depth map of a query image or video frame directly from a repository of 3D images (image + depth pairs or stereo pairs) using a nearest-neighbor regression type idea [1]. This method is built upon an assumption and a key observation. The observation is that among millions of 3D images that are available on-line, there exist many whose 3D content exactly matches that of a 2D input (query). And the assumption is that two images which are photo metrically similar also have similar 3D structure (depth). This is reasonable because photometric properties are often correlated and connected with 3D content of an image like depth and disparity. Edges of a depth map almost coincide with photometric edges.

1. Algorithm

Let us explain with an example. Given a monocular query image Q, assumed to be the left image of a stereopair that we wish to compute, we believe on the above observation and assumption that the entire depth field from a repository of 3D images I and render a stereopair. The steps of the process are as follows:

- Searching and finding the representative depth fields: In this step, we are going to find k 3D images in the repository I that have most similar depth to the query image, for example by performing a k nearest-neighbor (kNN) search using a metric based on photometric properties.
- Depth fusion to combine the depth fields: Here we combine the k representative depth fields. We can fuse the depth fields together by means of median filtering across depth fields.
- Depth smoothing by cross bilateral filter: We can process on the fused depth field in order to remove the spurious variations, along keeping with the discontinuities in depth. For example, by method of cross-bilateral filtering.
- Stereo rendering: In this step, we generate the right image of a fictitious stereopair using the monocular query image Q, and the smoothed depth field followed by suitable processing of occlusions and newly-exposed areas.
- Data hiding by steganography: After creating the converted image, we can put it into the steganograhic process for the data hiding in 3D image. Then by using extraction algorithm, we can extract the hidden data out of the 3D image

All the specific details of these steps depend on the type of 3D images which are in the repository. The above steps apply directly to 3D images represented as an image and depth pair. But the disparity field needs to be computed first in the case of stereo pair. So it should be done for each left or right pair of image. After that, each of the disparity field can be converted to a depth map. For example, under a parallel camera geometry assumption, along with fusion and smoothing takes place in the space of depths. The fusion and smoothing can take place alternatively in the space of disparities without make a change it into the depth. The final disparity can be used for right-image rendering.

Below figure shows the block diagram of our approach. The sections below in block diagram provide a describes each step and some high-level mathematical detail also. In these sections, Q_R is the right image which is being

DOI 10.17148/IJARCCE.2015.4707

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

sought for each query image Q, while d_Q is the query depth needed to numerically evaluate the performance of a depth computation. And we also assume that a 3D dataset I which is available by means of Kinect-based capture or disparity computation. The goal is to compute a depth estimate d and then a right-image estimate Q_R given a 2D query image Q and the 3D dataset I.

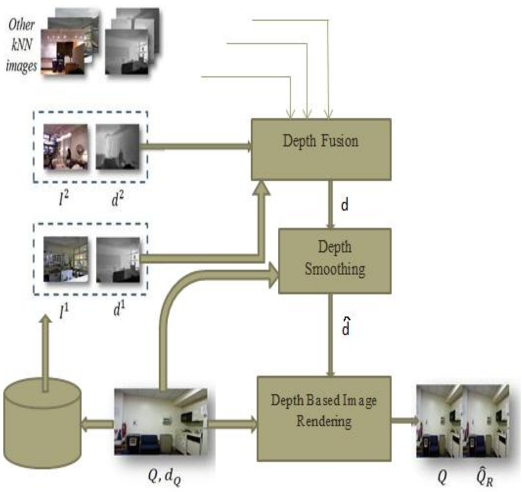

Figure 1: Block diagram of the overall algorithm: each block describes algorithmic details

2. Algorithm Description

- KNN Search

There exist two types of images in a large 3D image repository. Those images that is relevant for determining 2D query image and then the images that are irrelevant for determining. Images that are not photo metrically similar to the 2D query need to be rejected because they are not useful for estimating depth (as per our assumption). Note that although we might miss some depth-relevant images, we are effectively controlling the number of irrelevant images. These irrelevant images could be more potentially harmful to the 2D to 3D conversion process. When the size of the repository is very large, the selection of a smaller subset of images provides the added practical benefit of computational tractability.

A method of selecting a useful subset of depth-relevant images from a large repository is to select only the k images that are closest to the query where closeness is measured by the use of some distance function capturing global image properties such as color, texture, edges etc. In this function, we use the Euclidean norm of the difference between histograms of oriented gradients (HOGs) [4] computed from two images. Each HOG consists of 144 real values, that is it consists of 4 × 4 blocks with 9 gradient direction bins. It can be efficiently computed.

The search returns an ordered list of pair of image and depth. These are from most to the least photometrically similar query. We discard all but the top k matches (KNNs) from this list. The average photometric similarity between a query and its kth nearest neighbor usually decays with the increasing k [1]. While for large databases, larger values of k may be appropriate, since there are many good matches, for smaller databases this may not be true. Therefore, a judicious selection of k is important. We discuss the choice of k in Section.

- Depth Fusion

Even if none of the NN image and depth pairs match the query Q accurately, the location of some objects like furniture and parts of the background like walls is quite consistent with those in the respective query. If a similar object like building, table etc appears at a similar location in many kNN images, it is likely to have such an object also appears in the query. Then the depth field being sought should reflect this. We can compute this depth field by applying the median operator across the kNN depths. Although these depths are overly smooth, they provide a globally correct, even though coarse, assignment of distances to various areas of the scene.

- Cross-Bilateral Filtering (CBF) of Depth

While the median-based fusion helps make depth more consistent globally. Even if there is edge misalignment between depth fields of the kNNs and the query image, the fused depth is overly smooth and locally inconsistent with the query 2D image. This, in turn, frequently results in the lack of edges in the fused depth where sharp object boundaries should occur and/or the lack of fused-depth smoothness where smooth depth is expected.

Here we apply cross-bilateral filtering (CBF) [2]. CBF is a variation of the bilateral filtering, which is an edge preserving image smoothing method. It applies anisotropic diffusion controlled by the local content of the image itself [5]. In CBF, the diffusion is not controlled by the local content of the image under smoothing but it is done by using an external input. We then apply CBF to fused depth given by d using the query image Q in order to control diffusion. This will allow us in achieving two goals simultaneously: alignment of the depth edges with those of the luminance Y in the query image Q and local noise/granularity suppression in fused depth given by d. This can be implemented as follows:

$$\boldsymbol{\hat{d}[x] = \frac{1}{\gamma[x]}\sum_{y} d[y]h_{\sigma_s}(x-y)h_{\sigma_e}(Y[x]-Y[y])},$$

$$\boldsymbol{\gamma[x] = \sum_{y} h_{\sigma_s}(x-y)h_{\sigma_e}(Y[x]-Y[Y])},$$

Where is the filtered depth field and h σ(x) = exp (−‖x‖2/2σ2)/2πσ2 is a Gaussian weighting function.

Note that the directional smoothing of d is controlled by the query image via the weight hσe (Y [x] − Y [y]). For large luminance discontinuities, the weight hσe (Y [x]−Y [y]) is small and thus the contribution of d[y] to the output is small.

- Stereo Rendering

In order to generate an estimate of the right image Q R from the monocular query Q [12], we have to find out a disparity δ from the estimated depth d. Assuming that the fictitious image pair (Q, Q R) was captured by parallel cameras with baseline B and focal length f, the disparity is simply δ[x , y] = B f /d[x], where x = [x , y] T . We

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

forward-project the 2D query Q to produce the right image:

$$\widehat{Q_R}[x + \delta[x, y], y] = Q[x, y]$$

While rounding the location coordinates (x + δ[x, y], y) to the nearest sampling grid point. We handle occlusions by depth ordering: if (xi + δ[xi , yi], yi) = (x j + δ[x j , yi], yi) for some i, j , we assign to the location (xi + δ[xi , yi], yi) in Q R RGB value from that location (xi , yi) in Q whose largest. In newly-exposed δ[xi , yi] is the largest. In newly-exposed areas, i.e., for x j such that no xi satisfies (x j ,yi) = (xi + δ[xi , yi], yi), we apply simple in painting using in paint_nansfrom Mat-lab Central. Applying a more advanced depth-based rendering method would only improve this step of the proposed 2D-to-3D conversion.

First of all we have to read the input image and finding its edge detected image to calculate and store its boundary points. This step is also included in our 2D to 3D conversion algorithm. Here we are using the Global nearest neighbor depth learning algorithm. It contains the different steps which have already described. After finding out the converted image, we put it into the steganograhic process for the data hiding in 3D image. Then by using our extraction algorithm, at last we extract the hidden data out of the 3D image.

B. Steganography in 3D Image

Steganography is hidden writing, It is the art of hiding a secret data in such a way that no one except the intended recipient knows the existence of data. Usually just a steganography process indicates process in X-Y axis only, so we can store a less number of data in a cover image. Here we are intending to use the converted 3D images for data hiding. The 3D images have XYZ planes so we can store data's in X-Y or Y-Z or Z-X plane. So it is possible to hide more number of data's in a single image using different planes. Our methode is reserving room before encryption with a traditional RDH algorithm. Here it is easy to reversibly embed data into the encrypted image.

Reversible data hiding (RDH) in images is a technique by using this we can losslessly recover the original cover image after the embedded message is extracted. This technique is widely used in military imagery, medical imagery and law forensics,etc. In these places there is no distortion of the original cover is allowed. As it is first introduced, RDH has attracted considerable research interest.

In practical aspect, many RDH techniques have emerged in recent years. In most of the cases,themethode is that we first extracting compressible features of original cover and then compressing them losslessly, auxiliary data can be embedded into the spare space which is saved. Another popular method is based on DE or difference expansion, in which the difference of each pixel group is expanded. For example, if multiplied by two, the least significant bits (LSBs) of the difference are all-zero and can be used for embedding messages. The other good strategy for RDH is histogram shift (HS), in which space is saved for data embedding by shifting the bins of histogram of gray values. The state of art methods usually joined together DE or HS to residuals of the images.

Recently, more and more attention is paid to reversible data hiding (RDH) in encrypted images. Reason is that it maintains the excellent property and quality that the original cover can be losslessly recovered after extracted the embedded data along with protecting the image content's confidentiality. In all previous methods for embedding data by reversibly vacating room from the encrypted images, there may be subject to some errors on data extraction and image restoration. In this paper, we propose a method by reserving room before encryption with a traditional RDH algorithm, so it is easy for the data hider to reversibly embed data in the encrypted image. Experiments show that the method can embed more than 10 times as large payloads for the same image quality as the previous methods, such as for PSNR =40dB. This method can achieve real reversibility, that means it can achieve data extraction and image recovery are free of any error.

In this method, a content owner encrypts the original image using a standard cipher with an encryption key. The content owner can hand over the data to a data hider after producing the encrypted image (e.g., a database manager). Then the data hider can embed some auxiliary data into the encrypted image by losslessly vacating some room according to a data hiding key. After that a receiver himself or an authorized third party can extract the embedded data with the data hiding key. And after that we can further recover the original image from the encrypted version according to the encryption key. All previous methods embed data by reversibly vacating room from the encrypted images, and it may lead to some errors on data extraction and image restoration. It is difficult for data hider to reversibly hide the data behind the image. If we reverse the room prior to image encryption at content owner side, then the data hiding tasks in encrypted images would be more natural and much easier which leads us to the novel framework, "reserving room before encryption (RRBE)". Obviously, standard RDH algorithms are the ideal operator for reserving room before encryption and can be easily applied to Framework RRBE to achieve better performance compared with techniques from Framework VRAE.

In this system it uses traditional RDH algorithm, and thus it is easy for the data hider to reversibly embed data in the encrypted image. Using this system data extraction and image recovery are free of any error. In this module it compares the histogram of the image before encryption and the image after encryption. In this module stores the username and password to order for authentication and also the key to decrypt the message from the image.

This method uses the peak point to hide messages. we show that this scheme uses a reversible hiding strategy to achieve large hiding capacity and keep distortion low through a joint imperceptibility and hiding capacity evaluation. In the application level, reversible data hiding can be used as a fragile invertible authentication method in a lossless manner. And it embeds an authentication code into a digital image.

Only an authenticated party could extract the embedded information and restore the marked image to its pristine state. The image is completely authentic only if the embedded authentication code matches the extracted information. By embedding the message that has a close relationship to the host image, without requiring extra support reversible data hiding provides a self authentication scheme.

Similarly, a normal steganography has less security comparing with 3D image data hiding. Since the data is stored in a single plane in normal steganography process, it can be easily handled by the attackers and can hack it. But when we implemented the steganography in 3D image, there are 3 planes so that easy attack is not possible. We can also hide our key in any one of these planes, this also increase message confidentiality.

For the conversion of video, the input video is converted into frames; normally we are used 24frames/sec or 15frames/sec. This frames are images, this images are convert into 3D images via our existing algorithm. Here video sequences are created by the help of CODEC, it is an internal algorithm for joining frames into video sequences .We got 3D Video after depth fusion and depth smoothing.

IV. EXPERIMENTAL RESULT

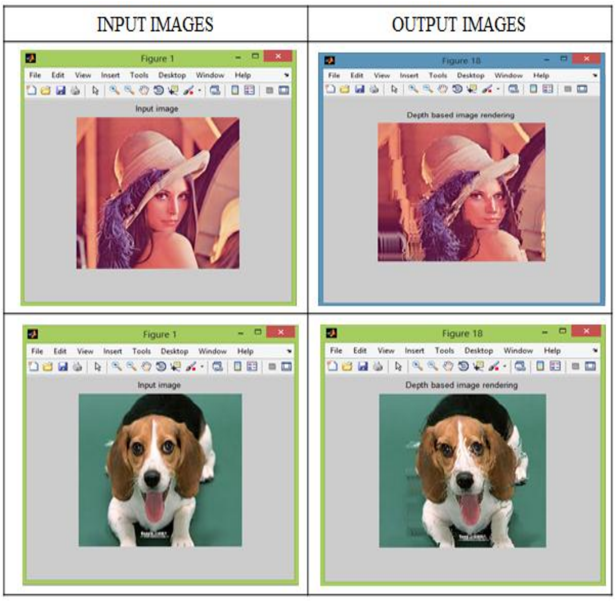

Figure 2: Showing the input and output images of 3D conversion

The proposed algorithm takes the 2D image and converts into 3D image by fusing together the left view and right view image. The left view and right view images are prepared using the depth values provided. The following table shows the images with different depth. The user shall fix the depth according to the requirement of image and basic feature of image.

The 2D to 3D converted images can be used in application level. Steganographic data hiding is trying to do in this converted 3D image. So it is possible to hide more number of data’s in a single image using different planes than in 2D image. We can also hide our key in any one of these 3 planes, this increase message confidentiality. By the steganographic data hiding using encryption before room leaving method, we can hide the data or message within the converted 3D image along with maintaining higher efficiency. And we can retrieve data from the 3D image without much distortion for the image so that we would get the non distorted 3D image after the data is successfully retrieved. For this recovery for the hidden data we have to use same encryption and decryption keyword for the security. So the MATLAB output screen display for the steganographic process will be as the following figure. This figure indicates just an example of screen display for the steganographic process in converted 3D image.

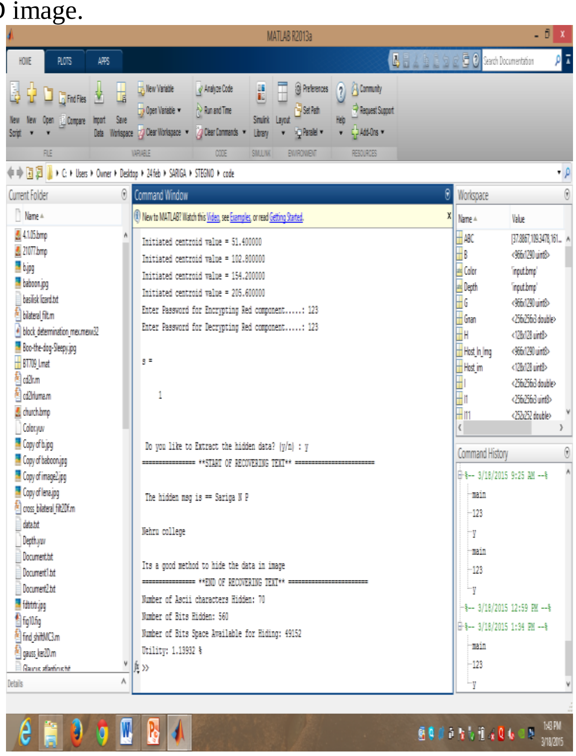

Figure 3: An example for output display of Steganographicdata hiding in MALAB

V. CONCLUSION

Here a new class of methods is proposed for 2D to 3D image conversion. This method is extremely fast when compared with the other existing algorithms. It performs at a fraction of CPU time.

The 2D to 3D converted images can be used in application level. Steganographic data hiding is trying to do in this converted 3D image. So it is possible to hide more number of data’s in a single image using different planes than in 2D image. We can hide our key in any one of these 3 planes, this also increase message confidentiality. We can also try for 2D to 3D video conversion along with steganography in 3D video.

ACKNOWLEDGMENT

My endeavour stands incomplete without dedicating my gratitude to everyone who has contributed a lot towards the successful completion of the project work. First of all,

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 7, July 2015

I offer my thanks to my parents for their blessings. I am indebted to God Almighty for blessing me with His grace and taking my endeavor to a successful culmination. I am very much grateful and thankful to our Principal **Prof. Dr. A. S. Varadarajan** for his guidance and support. I express my gratitude and heartfelt thanks to the Dean of Department of Electronics & Communication Engineering, **Prof. H. S. Divakara Murthy**, for giving directions and providing me with an opportunity to undertake this project. I would like to place on record my deep sense of gratitude to our Project coordinator **Mr. Murughanadham C,** Assistant Professor of Electronics & Communication Engineering Department for his valuable support and guidance throughout the course of the project. I would also like to extend my sincere gratitude and heartfelt thanks to my Internal Project Guide **Mrs. SajithaA S**, Assistant Professor of Department of Electronics Communication Engineering, for his support and guidance. I would like to thank all the faculties of NCERC for the help they have extended. Last, but not the least I thank my friends and all my well-wishers who supported me directly and indirectly, encouraged me and gave me the motivation to complete the project work.

REFERENCES

[1] J.Konard, Meng Wang, PrakashIshwar, Chen Wu, and D. Mukherjee, Learning-Based Automatic 2D-to-3D Image and Video Conversion, IEEE Trans., vol.22, no.9, Image Process., pp. 3485-3496, Sep. 2013

[2] L. Angot, W.-J. Huang, and K.-C. Liu, ÒA 2D to 3D video and image conversion technique based on a bilateral Plter,Ó Proc. SPIE, vol. 7526, 75260D, Feb. 2010.

[3] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, ÒHigh accuracy optical ßow estimation based on a theory for warping,Ó in Proc. Eur. Conf. Comput. Vis., 2004, pp. 25-36.

[4] N. Dalal and B. T0riggs, ÒHistograms of oriented gradients for human detection,Ó in Pr oc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2005, pp. 886-893.

[5] F. Durand and J. Dorsey, ÒFast bilateral filtering for the display of high-dynamic-range images,Ó ACM Trans. Graph., vol. 21, pp. 257-266, Jul. 2002.

[6] M. Grundmann, V. Kwatra, and I. Essa, ÒAuto-directed video stabiliza-tion with robust L1 optimal camera paths,Ó in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2011, pp. 225-232.

[7] M. Guttmann, L. Wolf, and D. Cohen-Or, ÒSemi-automatic stereo extraction from video footage,Ó in Proc. IEEE Int. Conf. Comput. Vis., Oct. 2009, pp. 136-142.

[8] K. Karsch, C. Liu, and S. B. Kang, ÒDepth extraction from video using non-parametric sampling,Ó in Proc. Eur. Conf. Comput. Vis., 2012, 775-788.

[9] J. Konrad, G. Brown, M. Wang, P. Ishwar, C. Wu, and D. Mukherjee, ÒAutomatic 2D-to-3D image conversion using 3D examples from the Internet,Ó Proc. SPIE, vol. 8288, p. 82880F, Jan. 2012.

[10] J. Konrad, M. Wang, and P. Ishwar, 2D-to-3D image conversion by learning depth from examples,Ó in Proc. IEEE Comput. Soc. CVPRW, Jun. 2012, pp. 16-22.

[11] M. Liao, J. Gao, R. Yang, and M. Gong, ÒVideostereolization: Combining motion analysis with user interaction,Ó IEEE Trans. Visualizat. Comput.Graph., vol. 18, no. 7, pp. 1079-1088, Jul. 2012.

[12] B. Liu, S. Gould, and D. Koller, ÒSingle image depth estimation from predicted semantic labels,Ó in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2010, pp. 1253-1260.

[13] R. Phan, R. Rzeszutek, and D. Androutsos, ÒSemi-automatic 2D to 3D image conversion using scale-space random walks and a graph cuts based depth prior,Ó in Proc. 18th IEEE Int. Conf. Image Process., Sep. 2011, pp. 865-868.

[14] A. Saxena, S. H. Chung, and A. Y. Ng, ÒLearning depth from single monocular images,Ó in Advances in Neural Information Processing Systems. Cambridge, MA, USA: MIT Press, 2005.

[15] A. Saxena, M. Sun, and A. Ng, ÒMake3D: Learning 3D scene structure from a single still image,Ó IEEE Trans. Pattern Anal. Mach. Intell., vol. 31, no. 5, pp. 824-840, May 2009.

BIOGRAPIES

Sariga N P pursuing M. Tech in Applied Electronics and Communication Systems at Nehru College of Engineering and Research Center, Thrissur, Kerala, India under University of Calicut. Participated and presented in National and International Conferences.

Sajitha A S, AMIE, M. Tech graduate specialized in Embedded Systems. Now, the Assistant Professor at Nehru College of Engineering and Research Center, Thrissur, Kerala, India.