RESEARCH ARTICLE OPEN ACCESS

Optimizing Monocular Cues for Depth Estimation from Outdoor Images

Aditya Venkatraman[1], Sheetal Mahadik[2]
[1](Department of Electronics &Telecommunication, ST Francis Institute of Technology, Mumbai, India)
[2] (Department of Electronics &Telecommunication, ST Francis Institute of Technology, Mumbai, India)

ABSTRACT
Depth Estimation poses various challenges and has wide range applications.Depth estimation or extraction refers to the set of techniques and algorithm's aiming to obtain distance of each and every pixel from the camera view point. In this paper, monocular cues are optimized for depth estimation from outdoor images.Experimental results of optimization of monocular cues shows that best performance is achieved in a monocular cue named haze on the basis of parameters such as RMS(root means square) error,total set of features and computation time.
Keywords–Depth estimation, haze, linear least squares problem, monocular cue, texture gradient

I. INTRODUCTION

People perceive depth remarkably well given just one image; we would like our computers to have a similar sense of depths in a scene. Upon seeing an image, a human has no difficulty understanding depth of every object from camera view point. However, learning depth remains extremely challenging for current computer vision systems. Depth estimation has important applications in robotics, scene understanding and 3-D reconstruction

Depth estimation or extraction refers to the set of techniques and algorithms aiming to obtain a representation of the spatial structure of a scene. In other terms, to obtain a measure of the distance of, ideally, each point of the scene. Depth estimation has continuously become an effort-taking subject in visual computer sciences. Conventionally, depths on a monocular image are estimated by using a laser scanner or a binocular stereo vision system. However, using a binocular stereo vision system requires adjustment on the camera taking a scanner picture, and using a laser scanner takes huge capitals as well, so both these apparatuses bring significant complexities. Therefore, some algorithms have been developed to process monocular cues in the picture for depth estimation. In related work, Michel's, Saxena & Ng [1] used supervised learning to estimate 1-D distances to obstacles, for the application of autonomously driving a remote control car

Three Dimensional (3D) imaging systems have attracted both commercial and scientific interest in different disciplines over the last few decades. While in the past years most of the research in the area of 3D imaging systems has concentrated on the stereoscopic technology, the fact that the viewer has to wear special headgear (e.g., stereoscopic glasses) in order to feel the 3D effect, has limited the acceptance and the application of them. The auto stereoscopic display systems are more comfortable for the viewer as they do not require the use of special glasses.

Most work on visual 3-D reconstruction has focused on binocular vision (stereopsis) [2] and on other algorithms that require multiple images, such as shape from shading [3] and depth from focus [4]. Depth estimation from a single monocular image is a difficult task, and re- quires that we take into account the global structure of the image, as well as use prior knowledge about the scene. Saxena's algorithm [5] generates depth map from monocular images. Gini & Marchi [6] used single-camera vision to drive an in-door robot, but relied heavily on known ground colors and textures. In order to avoid drawbacks of binocular cues and other depth estimation methods which require cues from two or more images, here, monocular cues are used for depth estimation. In this paper depth estimated using various monocular cues are compared and a monocular cue is selected based on parameters such as RMS (root mean square) errors, computation time and set of features.

II. METHODOLOGY

Depth estimation from monocular cues consists of three basic steps. Initially a set of images and their corresponding depth maps are gathered. Then suitable features are extracted from the images. Based on the features and the ground truth depth maps learning is done using supervised learning algorithm. The depths of new images are predicted from the learnt algorithm.Fig.1 indicates the block diagram of algorithm for comparison of different monocular cues.

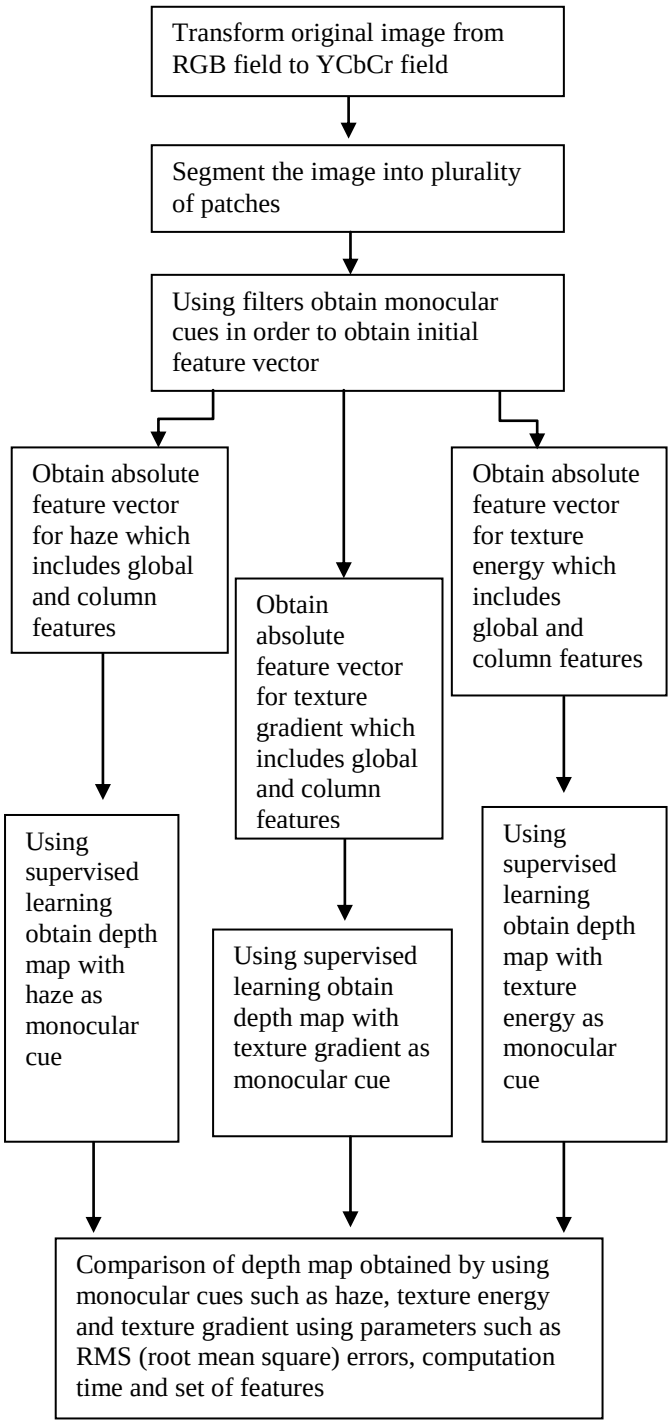

Figure 1: Block diagram of algorithm for comparison of different monocular cues

There are different monocular cues such as texture variations, texture gradients, interposition, occlusion, known object sizes, light and shading, haze, defocus etc. which can be used for depth estimation .The monocular cues used in this paper are haze, texture gradient and texture energy as these cues are present in most images .Many objects texture appear different depending on their distances from the camera view point which help in indicating depth. Texture gradients, which capture the distribution of the direction of edges, also help to indicate depth. Haze is another cue resulting from atmospheric light scattering.

Most of the monocular cues are global properties of an image and only little information can be inferred from small patches. For example, occlusion cannot be determined if we look at just a small portion of an occluded object. Although local information such as variation in texture and color of a patch can give some information about its depth, these are insufficient to determine depth accurately and thus global properties have to be used. For example, just by looking at a blue patch it is difficult to tell whether this patch is of a sky or a part of a blue object. Due to these difficulties, one needs to look at both the local and global properties of an image to determine depth. Thus local and global features are used to determine depth.

The local as well as global features are used in a supervised learning algorithm which predicts depth map as a function of image.

The detailed explanation of feature calculation and Learning is done in section 2.1, 2.2 and 2.3.

2.1 FEATURE VECTOR

The entire image is initially divided into small rectangular patches which are arranged in a uniform grid, and a single depth value for each patch is estimated. Absolute depth features are used to determine absolute depth of a patch which captures local feature processing (absolute features). Three types of monocular cues are selected: texture variations, texture gradients and haze. Texture information is mostly contained within the image intensity channel so Laws' mask is applied to this channel, to compute the texture energy. Haze is reflected in the low frequency information in the color channels, and is captured by applying a local averaging filter to the color channels. To compute an estimate of texture gradient, the intensity channel is convolved with Nevatia Babu filters or six oriented edge filters.

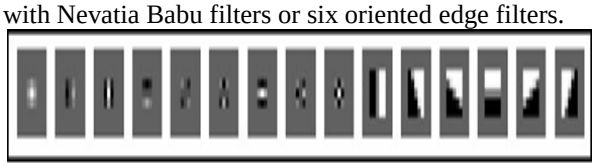

Figure 2: Filters used for depth estimation wherein, the first nine filters indicate Law's 3X3 masks and the last 6 filters are edge detectors placed at thirty degree intervals.

2.2 Absolute feature vector

Haze can be obtained by using local averaging filter (Law's mask filter) which is convolved with the color channels of an image. To compute an estimate of texture gradient, the intensity channel is convolved with Nevatia Babu filters which are six oriented edge filters. In the same way since texture information is mostly contained within the image intensity channel, nine Laws' masks are applied to this channel to compute the texture energy.

For a patch i in an image I(x,y) the outputs of local averaging filter, edge detection filters and Law's mask filters are used as:

$$E_i(n) = \sum_{(x,y) \in patch(i)} |I(x,y) * F_n(x,y)|^k \quad (1)$$

Where for $F_n(x, y)$ indicates the filter used which is local averaging filter for haze determination, edge filters for texture gradient determination and Law's filters for texture energy determination. Here n indicates the number of filters used, where, n=1 for haze determination, n=1..,..,6 for texture gradient determination and n=1,….,9 for texture energy determination. Here k = {1, 2} gives the sum absolute energy and sum squared energy respectively. Thus an initial feature vector of dimension 4 is obtained for Haze. An initial feature vector of dimension 12 is obtained for texture gradient and an initial feature vector of dimension 18 is obtained for texture energy.

With these filters local image features for a patch is obtained. But to obtain depth of a patch, local image features centered on the patch are insufficient, and more global properties of the image have to be used. Image features extracted at multiple image resolutions are used for this very purpose. Objects at different depths exhibit very different behaviors at different resolutions, and using multi-scale features (scale 1, scale 3, and scale 9) allows us to capture these variations. Computing features at multiple spatial scales also helps to account for different relative sizes of objects. A closer object appears larger in the image, and hence will be captured in the larger scale features. The same object when far away will be small and hence be captured in the small scale features. To capture additional global features, the features used to predict the depth of a particular patch are computed from that patch as well as the four neighboring patches which is repeated at each of the three scales, so that the feature vector of a patch includes features of its immediate neighbors, its neighbors at a larger spatial scale, and again its neighbors at an even larger spatial scale. Along with local and global features, many structures found in outdoor scenes show vertical structure so, additional summary features of the column that the patch lies in, are added to the features of a patch.

For each patch, after including features from itself and its four neighbors at 3 scales, and summary features for its four column patches, absolute feature vector for haze is obtained which is 19*4=76 dimensional.

Absolute feature vector for texture gradient is 19*12=228 dimensional and absolute feature vector for texture energy is 19*18=342 dimensional

2.3 SUPERVISED LEARNING

A change in depth along the row of any image as compared to same along the columns is very less. This is clearly evident in outdoor images since depth along the column is till infinity as the outdoor scene is unbounded due to the presence of sky. Since depth is estimated for each patch in an image, feature is calculated for each patch whereas learning done for each row as changes along the row is very less. Linear least squares method is used for learning whose equation is given by:

$$\Theta_r = \min (\sum_{i=1 to N} (d_i - x_i^T \Theta_r)^2) \quad (2)$$

In equation (2), N represents the total number of patches in the image. Here di is the ground truth depth map for patch i, x_i is the absolute feature vector for patch i. Here Θr, where r is the row of an image, is estimated using linear least squares problem

III. EXPERIMENTS

3.1 DATA

The data set is available online (http://make3d.cs.cornell.edu/data.html) which consists of 400 images and their corresponding ground truth depth maps which includes real world set of images of forests, campus areas and roadside areas.

3.2 RESULTS

The comparison of monocular cues is done on real-world test-set images of forests (containing trees, bushes, etc.), campus areas (buildings, trees and roads). The algorithm was trained on a training set comprising images from all of these environments. Here 300 images are used for training and the rest 100 images are used for testing.

TABLE 1 shows the comparison of different monocular cues based on RMS (root mean square) errors in various environments such as campus, forest and areas which include both campus and forest. The result on the test set shows that Haze has the least RMS error with an average error of in all the environments tested.

TABLE 2 shows the comparison of monocular cues based on computation time ,set of features used and average RMS error .It can be seen from that monocular cue named haze uses only 76 features as compared to texture gradient and texture energy which use 228 and 342 features respectively. Since haze uses less features, the total computation time of a depth map using haze is 9.8sec as compared to a total computation time of 29.4 sec and 44.1 using texture gradient and texture energy .Even though haze uses less features its average RMS is errors less than texture gradient and texture energy.

Table1: RMS errors of monocular cues when tested on different environments

Monocular cues	**Forest**	**Campus**	**Forest&campus (combined)**
Haze	0.774	0.824	0.853
Texture energy	0.886	0.840	0.876
Texture gradient	0.894	0.889	0.894

Table2: Comparison of monocular cues based on different parameters

Monocular cues	**Average RMS error**	**Computation time(sec)**	**Set of features**
Haze	0.817	9.89	76
Texture energy	0.867	29.4	228
Texture gradient	0.892	44.1	342

Figure 3: Original image (forest)

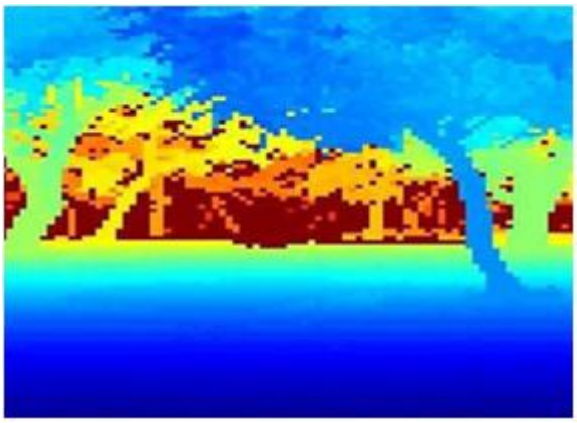

Figure 3.1: Ground truth depth map (forest)

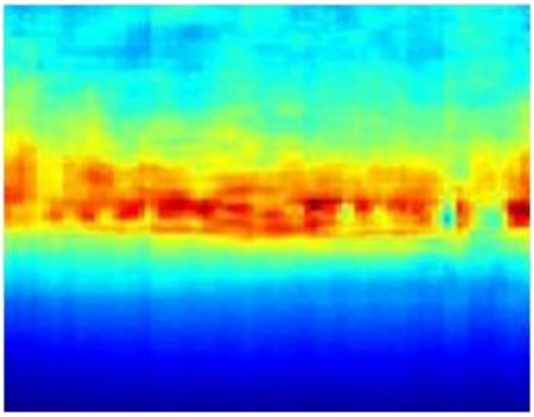

Figure 3.2: Depth map obtained using haze as monocular cue (forest)

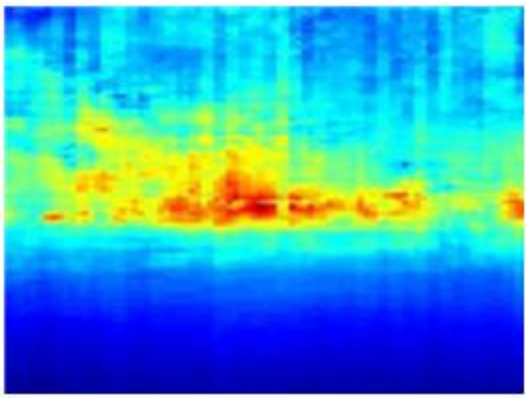

Figure 3.3: Depth map obtained using texture energy as monocular cue (forest)

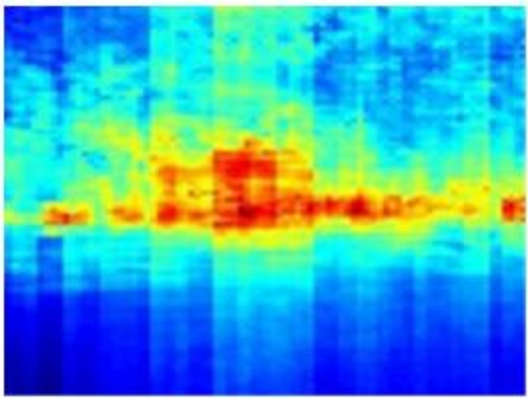

Figure 3.4: Depth map obtained using texture gradient as monocular cue (forest)

Figure 4: Original image (combined)

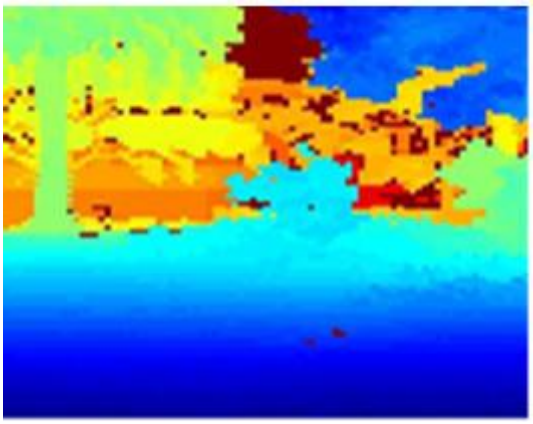

Figure 4.1: Ground truth depth map (combined)

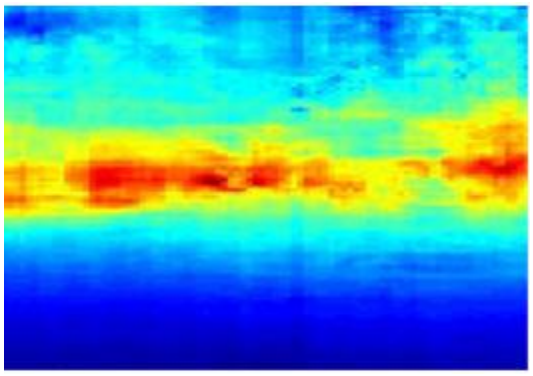

Figure 4.2: Depth map obtained using haze as monocular cue (combined)

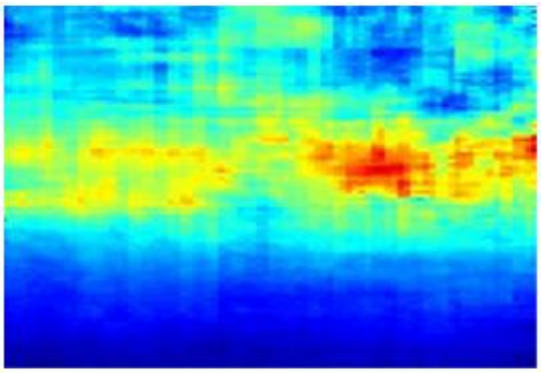

Figure 4.3: Depth map obtained using texture energy as a monocular cue (combined)

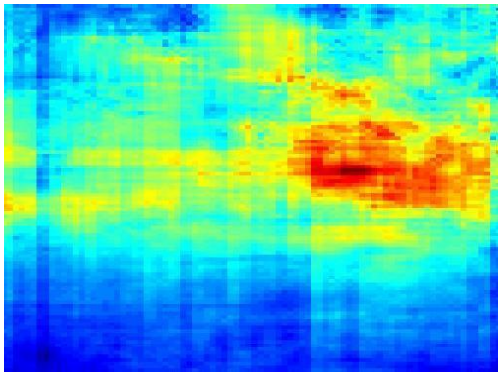

Figure 4.4: Depth map obtained using texture gradient as monocular cue (combined)

Figure 5: Original image (campus)

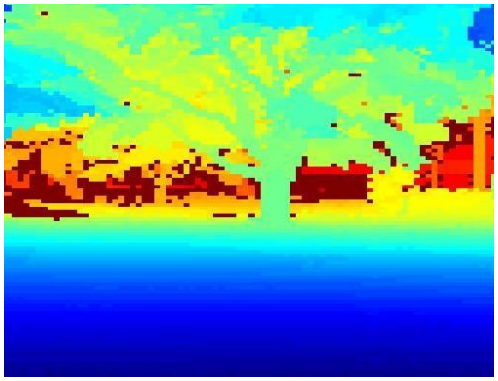

Figure 5.1: Original depth map (campus)

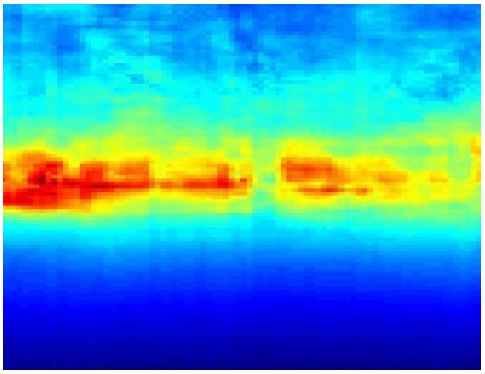

Figure 5.2: Depth map obtained using haze (campus)

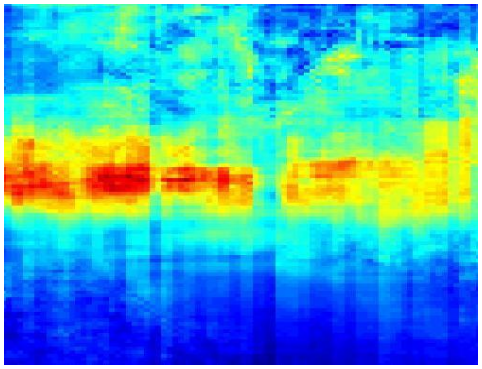

Figure 5.3: Depth map obtained using texture energy as monocular cue (campus)

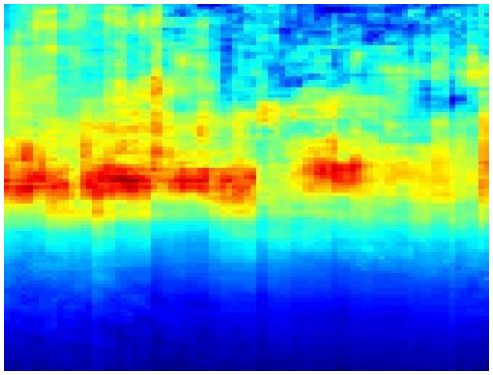

Figure 5.4: Depth map using texture gradient as monocular cue (campus)

IV. CONCLUSION

A detailed comparison of different monocular cues such as texture gradient, haze and texture energy in terms of RMS values, computation time and set of features respectively is done. In order to obtain a depth map using each of these monocular cues, local and global features are used. Depth map predicted using each of the monocular cues such as texture gradient, texture energy, haze are compared with the ground truth depth map and it is found with the implementation algorithm that haze has the least mean square error and hence it can be used as better monocular cue as compared to other monocular cues for depth estimation. Also haze uses very less set of features as compared to texture gradient and texture energy because of which feature optimization is achieved and computation time along with complexity is reduced. Hence use of Haze as monocular cue improves accuracy and provides feature optimization.

REFERENCES

[1] Jeff Michels, Ashutosh Saxena and A.Y. Ng. "High speed obstacle avoidance using monocular vision", In *Proceedings of the Twenty First International Conference on Machine Learning* (ICML), 2005.

[2] D.Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int'l Journal of Computer Vision*, 47:7–42, 2002

[3] M. Shao, T. Simchony, and R. Chellappa. New algorithms from reconstruction of a 3-d depth map from one or more images. In *Proc IEEE CVPR*, 1988.

[4] S.Das and N. Ahuja. Performance analysis of stereo, vergence, and focus as depth cues for active vision. *IEEE Trans Pattern Analysis& Machine Intelligence*, 17:1213–1219, 1995.

[5] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng. "Learning depth from single monocular images", In *NIPS 18*, 2006.

[6] G. Gini and A. Marchi. Indoor robot navigation with single camera vision. In *PRIS*, 2002.