Generative adversarial networks for depth map estimation from monocular images

Marcell Balogh
Technical Faculty of IT and Design
Department of Architecture, Design, and Media Technology
Aalborg University, Rendsburggade 14, 9000 Aalborg, Denmark
mbalog14@student.aau.dk

***Abstract*—Depth information is essential to determining geometric relations between objects, in particular, for obstacle avoidance. Unfortunately, depth-capable sensors are not as accessible as traditional RGB cameras, which limits the availability of depth-related cues. In this work, linear gray-level mapping methods are compared for depth estimation from images captured with monocular cameras. The proposed methods rely on processing of (i) a 3-channel RGB image, (ii) a gray-scale image with histogram equalization, and (iii) a gray-scale image with contrast limited adaptive histogram equalization (CLAHE), and map the respective inputs to an estimated depth map representation through conditional generative adversarial networks (cGANs).**

I. Introduction

Obstacle avoidance is one of the essential technologies in local path planning and one of the critical technologies that guarantees human and robot safety. 3D camera, LIDAR, SONAR and stereo camera sensors are widely used for depth estimation and hence these can be potentially used for obstacle avoidance. However, these sophisticated sensors are expensive and add unnecessary burden to a robot in terms of e.g. consumption of power or extra weights. However, monocular cameras are essential for every robot application while it is cheap and lightweight. Conversely, this paper addresses the topic of depth map estimation from a single monocular image using deep generative neural networks. This paper, in addition, delineates towards experimental investigation of the hypothesis that conditional generative adversarial networks can be improved by image processing techniques such as histogram equalization and contrast limited adaptive histogram equalization.

II. Implementation

A. Depth Maps

There is a wide selection of devices available in the market to produce depth maps along visual imagery. Depth map is a gray-scale image (0 to 255) in which each pixel value is the estimated distance from the camera to a surface and where nearer surfaces are darker and further surfaces are lighter. A depth camera, such as Microsoft Kinect (1), Intel Real Sense and Asus Xtion Pro use projected-light or also known as structured-light approach that combines the projection of a light (infrared) pattern with a standard 2D camera and that measure depth via triangulation. A similar technology is called

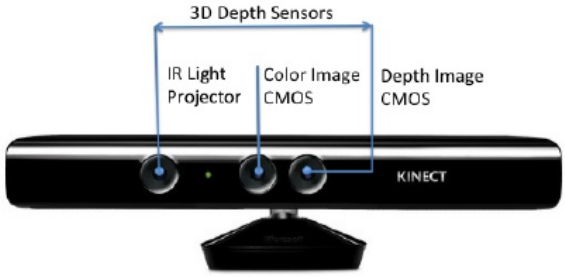


Fig. 1: Microsoft Kinect camera sensors. Source: https://www.researchgate.net/figure/Microsoft-Kinect-camera-sensorsfig1261987214.

time-of-flight that measures depth by estimating the time delay from light emission to light detection. However, these devices have some disadvantages such as scattered light that occurs due to unwanted reflections. Bright surfaces located very close to the camera quickly scatter too much light into the lens and create artifacts. Additionally, time-of-flight

distance measurements, for example, requires light that has been reflected just once. Light that is reflected multiple times distorts the measurement. Multiple reflections are typically produced by corners and concave shapes. Nevertheless, ambient light or environmental light such as sunlight makes outdoor use difficult. The high intensity of sunlight causes a quick saturation of the sensor pixels, thus, the actual reflected light from the light source (infrared) cannot be detected. Stereo or binocular vision is a branch of computer vision that extracts three-dimensional information out of two different images of the same subject. Conceptually, it traces imaginary lines from the camera to each object in the image, then does the same on the second image, and calculates the distance of objects based on the intersection of the lines corresponding to the same object. Stereo vision outputs a so-called disparity map, which is, just like a depth map, a grayscale image in which each pixel value is the stereo disparity of a surface. Nearby objects exhibit greater stereo disparity than far-off objects. Thus, nearby objects appear brighter in a disparity map. The quality of the computed disparity mainly depends on the appearance of the different objects that compose the scene. Highly textured regions tend to produce more accurate disparity estimates since they can be non-ambiguously matched. Also, a larger baseline increases the range of detectable depth values. However, enlarging the baseline also makes disparity computation more complex and less reliable.

B. Conditional Generative Adversarial Network

Recent studies [1] [2] showed that there is a deep learning technique to produce visual depth information using standard monocular camera only. The proposed technology uses conditional generative adversarial networks (GANs) [3] which is a machine learning approach for image-to-image translations. Conditional GANs are a type of generative adversarial network that are conditioned on some extra information which might be any kind of data, such as a class label, integer data or an image. There are two key components of GANs: the discriminator and generator. Generator is the generator model itself that takes a probability distribution (random noise) as input and tries to generate a realistic output. Discriminator takes two alternative inputs: the real images of the training dataset and the generated fake samples from the generator. It tries to determine whether the input image comes from the real images or the generated ones. Figure 2 shows an illustration of these two networks in action for depth map estimation. For depth map estimation, the process is

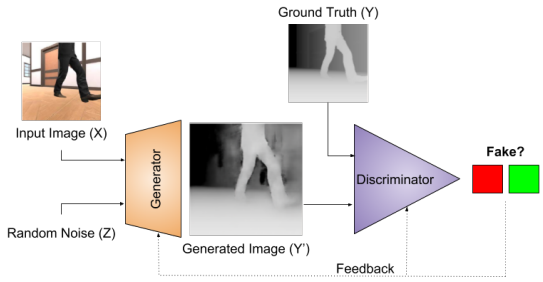


Fig. 2: High level representation of the conditional GANs architecture.

the following: GANs are given by RGB images (X) and the corresponding ground truth depth maps (Y) to train the generator and discriminator networks that generates depth maps (Y) on unseen visual data from random noise (Z). The two networks are trained together as a system. On the other hand, the discriminator tries to get better at distinguish between the real and fake images while the generator tries to output more realistic images so it could deceive the discriminator into thinking that the generated image is real.

C. Gray Level Mapping

Even though generative networks work efficiently, image processing techniques might helps to enhance depth maps estimation. A related work, for example, proposes [1] that gamma adjustment improve accuracy of the model. However, this project is motivated to find solution to prove the model by comparing gray level mapping and histogram methods. These methods are implemented through point processing which is a pixel-wise operation that calculates the new value of a pixel on each of the color channels in g(x, y) based on the value of the pixel in the same position in f(x, y). The values of a pixel's neighbors in f(x, y) have no effect whatsoever, hence the name point processing. Histogram is a graphical representation of the frequencies of events. However, image histogram is a tool to automatically access

whether an image is too dark, too bright or has too low or high contrast, and automatically correct the image using gray level mapping. Humans cannot tell the difference between grey level values too close to each other, thus these values could be spreading out what is called histogram stretching or equalization. Consider an image whose pixel values are confined to some specific range of values only. For example, brighter image will have all pixels confined to high values. But a good image will have pixels from all regions of the image. So the solution is to stretch its histogram to either ends, we can achieve this by mapping the leftmost nonzero bin in the histogram to 0 and the right-most non-zero bin to 255 (Figure 3). This normally improves the contrast of the image.

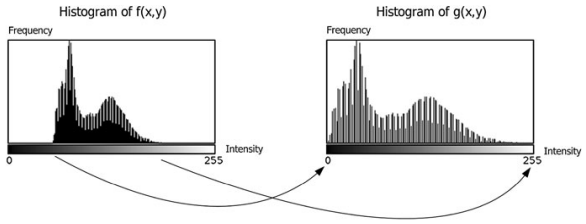


Fig. 3: Histogram Stretching. Source: http://what-when-how.com/introduction-to-video-and-image-processing/point-processing-introduction-to-video-and-image-processing-part-2/

Gray level mapping can be either linear or non-linear. Linear one manipulates the brightness

$$g(x, y) = f(x, y) + b \tag{1}$$

or the contrast of the image

$$g(x, y) = a * f(x, y) + b \tag{2}$$

while non-linear version, for example, adjust its gamma level.

$$g(x, y) = f(x, y)^{\gamma} \tag{3}$$

If all pixel values are increased (b>0), hence the image becomes brighter. This results in two things, firstly no pixel will be completely dark in the output and secondly some pixels will have a value above 255 in the output image. The latter is not good due to the upper limit of an 8-bit image and therefore all pixels above 255 are set to 255. When (b<0) the image becomes darker and some pixels will have negative values and therefore all pixels under 0 are set to 0. The contrast of an image is a matter of how different gray-level values are: if two pixels next to each other with close values then it is low contrast and if the pixels have a grater difference then it is high contrast. In this project two image processing methods were used and implemented in OpenCV library, histogram equalization and contrast limited adaptive histogram equalization (CLAHE).

D. Dataset Generation

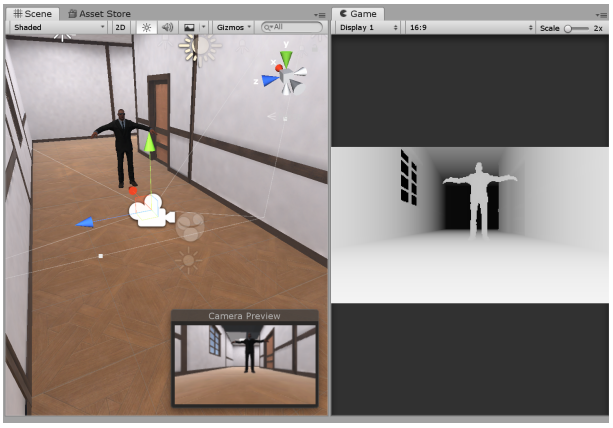


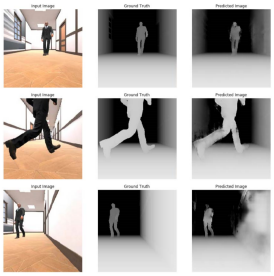
Fig. 4: The setup for dataset generation in Unity3D.

The depth maps for the datasets were created in simulated environment using a z-buffer shader in Unity3D. The environment consists of a UV textured 3D mesh corridor and an animated humanoid character. The Unity3D setup can be seen in Figure 4. The saved snapshots were cropped into 256x256 frames then the RGB images and depth maps were combined into 512x256 images to handle each sample easily during training. There are three datasets: one with histogram equalization, another with contrast limited adaptive histogram equalization and one without any image processing. In total, 56 pictures were generated, 50 for the train set and six for the test set.

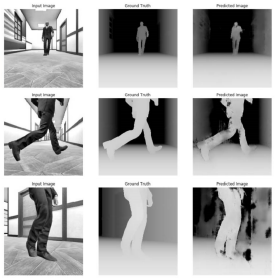
III. Evaluation

A. Evaluation Metrics

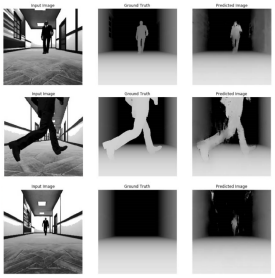
Two error-based evaluation metrics were used to evaluate the quality of the depth map reconstruction: root mean-squared error (RMSE) between the recon-



(a) Normal testset.



(b) CLAHE testset.



(c) Histogram equalization testset.

Fig. 5: Result samples of the depth map reconstructions.

structed single-channel depth map y' and the ground truth y is computed via:

$$RMSE(y, y') = (\frac{1}{n}\sum_{i=0}^{n}(y'_i - y_i)^2)^{\frac{1}{2}} \quad (4)$$

and normalized root mean-squared error (NRMSE), which is dividing the RMSE by the difference between the global maximum and the global minimum of the image pair and computed via:

$$NRMSE(y, y') = \frac{RMSE}{max(y' \oplus y)_i - min(y' \oplus y)_i} \quad (5)$$

where $\oplus$ is the concatenation operator.

B. Training Results

The training was performed on an Intel Core i7-7700K CPU and took 6.5 with the normal dataset of 400 epochs, 3.5 hours with CLAHE dataset of 200 epochs and 3.6 hours with histogram equalization of 200 epochs. The last two models with preprocessed data were trained with half amount of epochs less than the first round due to that the loss is likely to stagnate after 200 epochs. By stopping the training earlier over-fitting could be avoided. The experiment run with the speed of approximately one minute per epoch. The results of the depth map reconstructions can be seen in Figure 5 and the error measurements listed in Table I.

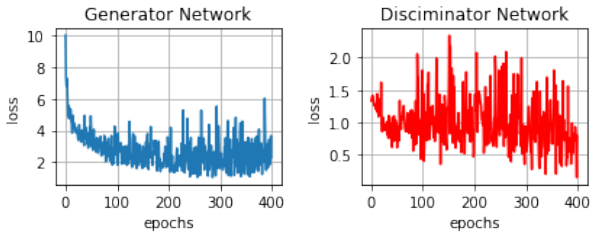


Fig. 6: Plots of the loss functions for both networks during training the normal dataset. At 200 epochs the loss stops to decrease.

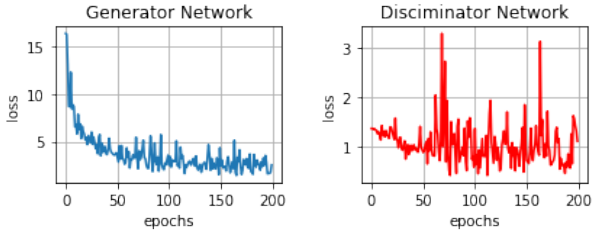


Fig. 7: Plots of the loss functions for both networks during training the CLAHE dataset.

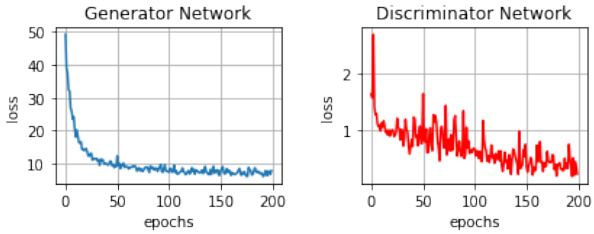


Fig. 8: Plots of the loss functions for both networks during training the histogram equalization dataset.

Adjustments	RMSE	NRMSE
None	**4.8019**	**1.3433**
CLAHE	4.8487	1.6268
HE	4.8413	1.53

TABLE I: The results of error-based evaluation metrics on each testset.

IV. Conclusion

The results clearly showed that histogram equalization converge the loss function smoother, however, the model without any image processing was trained faster than the other two conditions. Statistically, based on the root mean-squared error and normalized root mean-squared error evaluation metrics, there is no significant difference between the three conditions (Figure 9). Hence, based on this and the related

```
> pairwise.t.test(error, group, p.adjust="none", pool.sd = T)

        Pairwise comparisons using t tests with pooled SD

data:  error and group

  a    b
b 0.97 -
c 0.96 0.99
```

Fig. 9: Pairwise t-test for mean comparison of the three conditions.

work [1], I can conclude that non-linear gray-level mapping such as gamma correction improves the model in contrast the linear methods. Another conclusion is that the model could be improved with more data, and by using GPU the training process could be accelerated.

References

[1] K. G. Lore, K. Reddy, M. Giering and E. A. Bernal, Generative adversarial networks for depth map estimation from RGB video, 2018

[2] A. Singla, S. Padakandla and S. Bhatnagar, Memory-based Deep Reinforcement Learning for Obstacle Avoidance in UAV with Limited Environment Knowledge, 2018

[3] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, Image-to-image translation with conditional adversarial networks, in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017, pp. 5967 5976.