UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Environment reconstruction on disparity images using surface features and Generative Adversarial Networks

Lucas Peres Nunes Matias

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Lucas Peres Nunes Matias

Environment reconstruction on disparity images using surface features and Generative Adversarial Networks

Dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP – in accordance with the requirements of the Computer and Mathematical Sciences Graduate Program, for the degree of Master in Science. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Denis Fernando Wolf

USP – São Carlos
February 2020

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

P437e Peres Nunes Matias, Lucas
Environment reconstruction on disparity images
using surface features and Generative Adversarial
Networks / Lucas Peres Nunes Matias; orientador
Denis Fernando Wolf. -- São Carlos, 2020.
66 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2020.

1. Image Inpainting. 2. GAN. 3. Depth
Estimation. 4. Object Removal. 5. Disparity Images.
I. Fernando Wolf, Denis, orient. II. Título.

Lucas Peres Nunes Matias

**Reconstrução do ambiente em imagens de disparidade
utilizando informações das superfícies e Redes Adversariais
Generativas**

Dissertação apresentada ao Instituto de Ciências
Matemáticas e de Computação – ICMC-USP,
como parte dos requisitos para obtenção do título
de Mestre em Ciências – Ciências de Computação e
Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e
Matemática Computacional

Orientador: Prof. Dr. Denis Fernando Wolf

USP – São Carlos
Fevereiro de 2020

ACKNOWLEDGEMENTS

To my parents Emanuel Franco Matias and Damaris Peres Nunes Matias, and my sister Agatha Peres Nunes Matias who always supported, encouraged me and worked hard so I could follow and achieve my dreams;

To my girlfriend Fernanda Rani Malta for the patience, comprehension, and support on the hard times that I had during this Journey;

To my laboratory friends who made easier the struggle during the times of study;

To my friends Caio Vaz, Denise Alves, Kennedy Rodrigues, Lucas Pinto Soares, Matheus Marcondes de Abreu and Renan Gomes Vieira who helped me during the hard times of stress to distract and enjoy the simple things of life;

To my advisors Prof. Dr. Denis Fernando Wolf and Prof. Dr. Jefferson Rodrigo de Souza who guided me during this work, helping me to improve, understand and identify my failures and success;

To CNPq (Conselho Nacional de Pesquisa e Desenvolvimento), for the scholarship and financial support during all this work, which made possible the development and conclusion of this work;

To the professors of the Graduate Program of the Institute of Mathematics and Computer Science for the knowledge and incentive during this Journey.

“Every grain of experience is food for the greedy growing soul of the artist.”
(Anthony Burgess)

RESUMO

MATIAS, L. P. N. **Reconstrução do ambiente em imagens de disparidade utilizando informações das superfícies e Redes Adversariais Generativas**. 2020. 66 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

O estudo e desenvolvimento de veículos autônomos vem se tornando cada vez mais relevante. Para que estes veículos possam trafegar em um ambiente urbano real é necessário alto grau de confiabilidade para garantir a segurança dos passageiros e dos pedestres. Para isso, os sensores e algoritmos utilizados para auxiliar na tomada de decisão durante a direção autônoma necessitam do máximo de informação disponível, para que a análise do ambiente seja o mais ampla possível. Assim como um motorista humano, o computador deve analisar o ambiente ao seu redor e avaliar as possíveis medidas a serem tomadas afim de alcançar o destino final do trajeto de modo seguro. Contudo, apesar dos sensores coletarem informações com alta precisão, métodos computacionais possuem uma desvantagem em relação à cognição humana. Um motorista humano pode analisar as informações do ambiente ao seu redor e deduzir informações oclusas, mais especificamente informações sobre o ambiente por de trás de diferentes objetos e estruturas. Para métodos computacionais, extrair esta informação omissa é um desafio. Trabalhos recentes na área de processamento de imagens propõe métodos para estimar a área por trás de regiões especificadas. Porém, estes métodos são aplicados a imagens RGB, onde o foco é um resultado visualmente satisfatório. Ao lidarmos com imagens de disparidade, que codificam dados de profundidade, é necessária uma estimativa coerente e precisa, uma vez que quaisquer ruídos na imagem, serão intensificados na reconstrução tridimensional e influenciarão na interpretação do ambiente ao redor pelos algoritmos de tomada de decisão. Neste trabalho lidamos com a hipótese de que, utilizando aspectos específicos dos dados de disparidade e profundidade como orientação para a estimativa de disparidade, é possível alcançar uma reconstrução coerente do ambiente por de trás de uma região demarcada. Os resultados apontam para a validação dessa hipótese, uma vez que alcançamos - ao fim deste trabalho - uma reconstrução contínua do ambiente com poucos ruídos.

Palavras-chave: Reconstrução de Imagens, GAN, Estimação de Profundidade, Remoção de Objetos, Imagens de Disparidade.

ABSTRACT

MATIAS, L. P. N. **Environment reconstruction on disparity images using surface features and Generative Adversarial Networks**. 2020. 66 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

The study and development of autonomous vehicles have become more relevant at each day. For the intelligent vehicle to be able to navigate through a real urban environment it is necessary a high degree of reliability to ensure the passenger and pedestrians safety. Therefore, sensors and algorithms used to help the decision making during the autonomous navigation need the maximum amount of information available, for the environment analysis to be most complete as possible. As an human driver, the computer should analyze the surrounding environment and evaluate the possible actions to execute in order to reach the final destination safely. However, despite the high precision data collected by the sensors, computational methods has a disadvantage when compared to the human cognition. A human driver can analyze the surrounding environment and deduce occluded information, more specifically information related to the environment behind objects and structures. For computational methods extract this missing data is a challenge. Recent works on image processing propose methods to estimate the area behind specified regions. Yet, those methods are applied on RGB images, where the focus is a visually satisfactory result. When dealing with disparity images, which codify depth data, it is necessary a coherent and precise estimation, since any noise on the image will be intensified in the tridimensional reconstruction, and will influence on the decision making algorithms environment interpretation. In this work we deal with the hypothesis of, by using specific disparity and depth features as guideline for the disparity image estimation, it is possible to achieve a coherent environment reconstruction of the area behind a masked region. Our results point out to this hypothesis validation, since we achieve - at the end of this work - a continuous environment reconstruction without significant noise.

Keywords: Image Inpainting, GAN, Depth Estimation, Object Removal, Disparity Images.

LIST OF FIGURES

Figure 1 – LiDAR data visualization.	26
Figure 2 – Color and disparity image estimated by the stereo matching algorithm. . . .	26
Figure 3 – LeNet architecture for handwriting digits recognition.	27
Figure 4 – GAN diagram where, given an input noise, data is generated and passed to the discriminator to distinguish between real and fake data.	28
Figure 5 – GAN and WGAN loss function gradient comparison. In blue, the real data and in green the fake data distribution. In red is displayed the cross-entropy function gradient, dividing real and fake data, where the vanishing gradient is highlighted. In light blue, the WGAN gradient is displayed, which is smoother every.	29
Figure 6 – WGAN gradient norm with different weight clipping and WGAN-GP. The different weight clipping changes the gradient norm drastically, behaving as a weight regulation, while the gradient penalty keeps the norm closer to a central value.	31
Figure 7 – Inpainting process for object removal. (a) The original image; (b) the image with the woman removed from the scene.	34
Figure 8 – Depth completion example. (a) Input depth maps where white regions are missing data; (b) the result from depth completion method; (c) difference between input and result image.	36
Figure 9 – Sparse Convolution results from Uhrig et al. (UHRIG <i>et al.</i> , 2017) compared with other methods.	36
Figure 10 – Encoder-decoder inpainting GAN.	40
Figure 11 – Generative Inpainting Network architecture. The coarse result is the output from the original encoder-decoder architecture, post-processed with the attention branch in the refinement network.	41
Figure 12 – Contextual attention network applied to RGB and disparity images, evidencing the reconstructed area.	42
Figure 13 – Relation between adjacent pixels and the surface normal vector calculation.	43
Figure 14 – Normal vectors estimation from color variation around x and y image axis. In green, the y parallel vector, in red, the x parallel vector and in blue the vector normal to the two vectors surface.	44
Figure 15 – Surface normals image generated by the generalized formulation from the disparity image.	45

Figure 16 – Kernel for surface normal estimation.	45
Figure 17 – Contextual Attention network diagram with surface normals estimation and Vectorial Loss.	46
Figure 18 – Contextual Attention branch diagram.	46
Figure 19 – Contextual Attention network diagram with surface normals estimation, Vectorial Loss, and Surface Attention adaptation.	48
Figure 20 – Our proposed GAN architecture with surface normals analysis in many of the data processing stages.	49
Figure 21 – Different scenes examples captured in the KITTI Dataset.	52
Figure 22 – CityScapes examples with the RGB image and depth maps partially overlayed, CityScapes instance labelling and different semantic segmentation approaches using their dataset.	52
Figure 23 – Networks disparity and surface reconstruction comparison. (a) Original RGB, disparity and surface images with the cars to be removed; (b) original network; (c) network with Vectorial Loss; (d) network with Vectorial Loss and Surface attention branch; (e) our proposed architecture with Vectorial Loss, Surface Attention branch and Surface Discriminator.	55
Figure 24 – Mean squared pixel error from the network with Vectorial Loss only and with Vectorial Loss and Surface Discrimination.	56
Figure 25 – Comparison between Contextual Attention network (YU <i>et al.</i> , 2018) depth reconstruction and ours. First column the RGB image; Second column the original disparity image; Third column the Contextual Attention reconstruction of the area behind the target objects; Last column our reconstruction.	58
Figure 26 – Comparison between Contextual Attention network (YU <i>et al.</i> , 2018) depth reconstruction and ours. First column the RGB image; Second column the original disparity image; Third column the Contextual Attention reconstruction of the area behind the target objects; Last column our reconstruction.	59
Figure 27 – Examples of object removal on different scenes with our proposed network.	60

LIST OF TABLES

Table 1 – Encoder-decoder and the Contextual Attention networks with and without the Vectorial Loss	54
Table 2 – Mean-Squared Error and Vectorial Error calculated from each network adaptation and original Contextual Attention network.	55
Table 3 – Data distribution distances between ground truth and reconstructed images for each applied adaptation. CA is the original Contextual Attention network (YU <i>et al.</i> , 2018); VL is the same network with our Vectorial Loss function; SA the network with Vectorial Loss and Surface Attention branch adaptation; Ours is our final proposed network with Vectorial Loss, Surface Attention branch and Surface Discrimination.	57

LIST OF ABBREVIATIONS AND ACRONYMS

CNN	Convolutional Neural Networks
CRF	Conditional Random-Field
DARPA	Defense Advanced Research Project Agency
ELAS	Efficient Large-Scale Stereo Matching
GANs	Generative Adversarial Networks
GPU	Graphics Processing Unit
ITS	Intelligent Transportation Systems
LiDAR	Light Detection and Ranging
MRF	Markov Random-Fields
MSE	Mean-Squared Error
RGB	Red Green Blue
VE	Vectorial Error
WGAN	Wasserstein GAN
WGAN	Wasserstein GAN
WGAN-GP	WGAN with Gradient Penalty

CONTENTS

1	INTRODUCTION	21
1.1	Objectives	23
1.2	Chapters Organization	24
2	THEORETICAL BACKGROUND	25
2.1	Depth Sensors	25
2.1.1	<i>Stereo Camera</i>	25
2.2	Convolutional Neural Networks	26
2.3	Generative Adversarial Networks	27
2.3.1	<i>Wasserstein GAN</i>	29
2.3.2	<i>Wasserstein GAN with Gradient Penalty</i>	30
2.4	Final Considerations	30
3	LITERATURE REVIEW	33
3.1	Image Inpainting	33
3.1.1	<i>Related Works</i>	33
3.2	Depth Completion	35
3.2.1	<i>Related Works</i>	35
3.3	Final Considerations	38
4	PROPOSED WORK	39
4.1	Inpainting GANs	39
4.1.1	<i>Color Inpainting in Disparity Data</i>	41
4.2	Proposed Architecture	41
4.2.1	<i>Depth Surfaces</i>	42
4.2.2	<i>Vectorial Loss</i>	45
4.2.3	<i>Surface Attention Branch</i>	46
4.2.4	<i>Surface Discrimination</i>	48
4.3	Final Considerations	48
5	RESULTS	51
5.1	Training Datasets	51
5.2	Results	53
5.2.1	<i>Vectorial Loss</i>	53

5.2.2	<i>Proposed Architecture</i>	54
5.2.3	<i>Statistical Distance Metrics</i>	56
5.3	Final Considerations	58
6	CONCLUSIONS	61
	BIBLIOGRAPHY	63

INTRODUCTION

Nowadays, investment in intelligent systems has been widely increased. Computers executing tasks faster, better, and cheaper than a human is an idea that interests most of the corporations. A challenging and exciting application is autonomous intelligent vehicles. This task is a critical one to be implemented by a computer since it involves human lives and its safety. To deploy a functional autonomous vehicle, it is necessary for high reliability on those systems, which depends on precise sensor data and high processing capacity.

In the last two decades, Intelligent Transportation Systems (ITS) have been widely studied and developed thanks to the funding from vehicles and transportation companies. This race for a final product to be commercialized increase the velocity on which this technology development advances. To support these advances, it is necessary more precise sensors and high computational processing power. Perception systems have to ensure reliable and accurate information about the environment to help the decision making, so the vehicle will be able to understand the trajectory to follow and their surroundings. For autonomous vehicle's vision systems, the collected data precision is critical since the trajectory, obstacle, and environment perception are deeply related to the passenger's and pedestrian's safety.

The milestone for the autonomous vehicles development was the Defense Advanced Research Project Agency (DARPA) Challenge. The first DARPA challenge occurred in 2004, where several research groups competed presenting their autonomous vehicles. The challenge was to finish a trajectory in the USA Mojave Desert successfully. However, in this challenge, none of the groups could finish the trajectory. Though in the next year (2005), this challenge was again proposed, and at this time, five teams could finish the trajectory. The first place was the Stanford Racing Team with their autonomous vehicle Stanley. Once completed this challenge, in 2007, a new one and more difficult was proposed, the DARPA Urban Challenge. The difficulty in this new challenge was the change of an unstructured environment to a complex structured environment. The Urban Challenge was held in a simulated city, where the vehicle should follow the road, identify and deal with different obstacles, and obey many traffic rules. Those tasks

were a challenge, especially for the perception systems. The simple perception system that once needed to identify the navigable area and any obstacle to avoid was replaced to a robust system where it was necessary to identify the road, obstacles different classes, and process this information on the decision-making system. Those new tasks increased the interest and study on perception systems, and more precise sensors were developed to deal with those complex scenarios.

As the demand for more robust perception systems increased, more semantic information was needed. To extract semantic data from the scene is necessary more than just depth information, given by Light Detection and Ranging (LiDAR) sensors. Red Green Blue (RGB) images can give more semantic information since it is possible to extract color, texture, edges, and shape from them. For this purpose, the use of stereo cameras started to be more relevant. Stereo cameras use multiple lenses, capturing images at the same time. Since those lenses have different positions, it is possible to use the multiple scene perspectives to estimate a depth value for each pixel. With this sensor, it is possible to extract depth data and semantic features related to the RGB image. Due to that, studies related to stereo images increased, focusing on improving depth estimation accuracy. Moreover, since it is an image-based sensor, traditional image processing algorithms can be applied to this data, increasing the possible applications for those sensors. Also, when comparing the data from stereo cameras and LiDAR sensors, the stereo cameras depth data are far denser than LiDAR data, though it lacks precision.

Alongside the ITS area, the image processing methods also advanced, which had a huge impact on many related research topics. A remarkable point for those fields was the introduction of the Convolutional Neural Networks (CNN). The first glances of the Convolutional Neural Networks (FUKUSHIMA, 1980; Lecun *et al.*, 1998), as known today, were set aside due to the high processing power necessary to fit a large number of weights on the convolutions. With the emerging of Graphics Processing Unit (GPU), it became possible to process more data in less time. Thanks to that, in 2012, AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) was presented, highlighting CNN architectures again. This paper proposed a complex neural network architecture based on convolutions; with this architecture, the network learns which feature is more relevant to complete the image processing task, abstracting the problem of defining filters to extract information from the input data. The notable results from those architectures encouraged the research on CNNs, quickly achieving significant advances in this area. The ITS field took advantage of those networks by increasing the accuracy of many semantic data extraction tasks. Thanks to those architectures new and more complex tasks emerged, e.g. semantic segmentation (BADRINARAYANAN; HANDA; CIPOLLA, 2015), instance segmentation (CORDTS *et al.*, 2016a) and pedestrian intention detection (Ridel *et al.*, 2018). The solution to those tasks aggregates to the vehicle perception system important information about the surroundings and specific semantic information about the obstacles in it. With this knowledge level about the obstacles, the decision-making system could interpret and make specific actions to deal with different scenarios. Using stereo cameras as input for those algorithms withdraws the

synchronization problem related to using multiple perception sensors since they can provide both depth and RGB information in the same frame.

For many of those semantic tasks, the obstacles are of interest, and the background environment has less relevance. However, for other tasks e.g. mapping (YANG; HUANG; SCHERER, 2017), curb detection (FERNÁNDEZ *et al.*, 2015), and instance segmentation (CORDTS *et al.*, 2016a), the environment is a fundamental part, while the obstacles information is irrelevant. For those environment focused tasks, the obstacles can be a problem since they hide a significant part of the environment. Behind an object, there is unknown information; due to the light physical properties, this region can not be seen, since there is one object between that area and the camera lenses. This kind of hidden information is called occlusion. When compared to a human, the computer fails to infer that occluded area. The human mind deduces the missing information based on earlier experiences and knowledge about the environment to guess what is behind an object. For computational methods, to achieve this guess, it would be a hard task due to the lack of previous knowledge about the environment. Berlincioni et al. (BECATTINI *et al.*, 2018) proposes the use of Generative Adversarial Networks (GANs) to remove objects from semantic segmented images and estimate the semantic structures behind these objects. This work can estimate hidden semantic information; however, it can not deal with occluded depth data, which is more complex. To achieve a realistic depth reconstruction, it would be necessary a precise estimation of the hidden area due to the depth data noise sensitivity. The challenge is to estimate a coherent reconstruction with the necessary precision to not spoil the tridimensional data integrity. With this reconstruction, the perception system would have a plausible supposition about the missing region, using this information to support different perception tasks.

1.1 Objectives

Many computer vision methods deal with image missing data estimation. Some methods focus on object removal and image reconstruction, e.g., inpainting methods. Some other approaches try to deal with different lack of information on depth maps, e.g., depth completion. The arduous task of applying RGB methods to disparity images is the noise sensitivity of depth data. The noises on RGB images are not problematic since it is focused on a global visually good result. However, on depth images, the pixel neighborhood directly influences the depth estimation, which means that any noise can be problematic to the tridimensional reconstruction.

Our goal is to reconstruct an area on depth maps estimating disparity data. To achieve a coherent and plausible reconstruction, it is necessary a smooth and precise estimation. With this plausible guess, we can specify and remove objects from disparity images, reconstructing the environment behind this area. With a robust method able to do so, we could have many possible applications and advancements in different ITS fields, e.g., object tracking, mapping, and semantic segmentation. On all those tasks, a plausible supposition about the area behind an

object would increase the information related to the occluded region, helping the prediction and classification of those methods. In this dissertation, we propose a method able to reconstruct a masked area on disparity images, based on its neighborhood data. Our work can support other tasks on ITS, increasing their information about the environment and, thereafter, their accuracy.

1.2 Chapters Organization

This dissertation is divided into six chapters. Chapter 1 contextualize our problem and discuss about our motivation. Chapter 2 introduce the theoretical background, defining specific image processing tasks, algorithms, and methods. Related works are discussed and presented in Chapter 3, here we explain the positive points and flaws of those methods when applied to our problem. In Chapter 4, we define our hypothesis and reason about our proposed methodology to achieve the problem solution. In chapter 5, we discuss ways to evaluate GAN generated images and argue about our qualitative and quantitative results. At last, in Chapter 6, we present our conclusion about this work and summarize our main contributions.

THEORETICAL BACKGROUND

This Chapter presents the terminology and theoretical background of our work. We discuss depth sensors, specifically Stereo Cameras and their algorithms for depth information estimation. Also, we present the problems when using stereo images and the flaws when collecting these data. Then, we briefly introduce CNNs and present the concept behind GANs and how they generate data. Thereafter, we discuss improved GANs models, specifically the Wasserstein GAN (WGAN), which is the basis of our approach.

2.1 Depth Sensors

Depth sensors are a fundamental part of the robotics and ITS field. Intelligent vehicles have to navigate in a complex scenario where a new obstacle can appear at any time. Since those vehicles navigate through the environment at high speed, any collision can be fatal. That is why it is necessary to collect reliable information from those sensors. LiDAR sensors can collect precise depth information at a high frequency, generating a point cloud with the distance of each point to the sensor (Figure 1). Still, the data from farther regions are too sparse, aggregating poor information about those areas. Stereo camera collected data are far denser, which provides more information about the environment. Also, stereo cameras collect RGB information; from these data, shape, texture, and edge information can be extracted. Although, stereo camera depth data are less precise than LiDAR sensors point clouds.

2.1.1 Stereo Camera

Stereo cameras are biologically inspired by the human eye. Using different lens perspectives and a stereo matching algorithm is possible to calculate a disparity value based on the difference between the camera lenses (Figure 3). With these disparity values and the camera configuration parameters, a depth value can be calculated for each disparity pixel, generating

Figure 1 – LiDAR data visualization.

a point cloud from RGB images. Therefore, we can assign a depth value for each RGB pixel without any synchronization between sensors.

Figure 2 – Color and disparity image estimated by the stereo matching algorithm.

Stereo matching algorithms extract and match features from the camera multiple frames. Based on the differences between the frames is calculated a disparity value for each pixel. For some pixels, a disparity value can not be calculated due to unmatched features in that region. Since the lenses have different perspectives, some regions that are captured by one lens are not captured on the others, thereat, for that specific area, there will be no matched features, and the disparity value will be equal to zero. Due to that, disparity images may have shadowed areas without depth information, which is also named occlusion; however, this occurs for a different reason. In our work, we deal only with information hidden by objects in the scene.

2.2 Convolutional Neural Networks

Convolutional Neural Networks are architectures focused on data feature extraction based on convolutional filters. Those networks are proposed to deal with image processing, modeling

the image filters kernels as trainable weights. In this architecture, the convolutions are fitted to extract specific features from the input data. Between the convolutional filters, the data can be redimensioned to accomplish different tasks, e.g., image recognition (RAWAT; WANG, 2017) and object detection (DRUZHKOVA; KUSTIKOVA, 2016). To correctly fit the convolutional weights, it is necessary a large amount of data, to extract the right features to deal with the proposed task.

Figure 3 – LeNet architecture for handwriting digits recognition.

Source: [Lecun et al. \(1998\)](#).

Due to the impressive results achieved with CNNs, many other approaches are proposed based on convolutional filters to deal with different research domains, e.g., Transfer Learning (TAN et al., 2018), Generative Adversarial Networks (Pan et al., 2019) and Convolutional Recurrent Neural Networks (LIPTON; BERKOWITZ; ELKAN, 2015). CNNs can extract sophisticated features from the input data to deal with different tasks, making those networks applicable in many research areas. A Comprehensive Review of CNNs can be found at (Aloysius; Geetha, 2017).

2.3 Generative Adversarial Networks

Generative Adversarial Networks are CNNs architectures focused on data generation. They were introduced by Goodfellow et al. in (GOODFELLOW et al., 2014). This network tries to replicate data distribution from a training dataset generating new instances (e.g., images, sound, points). To do so, these networks abstract the qualitative evaluation by training a second network to distinguish between real and fake data (Figure 4). The *Adversarial* suggests that both network - generator and discriminator - will compete to improve each other. As the generator network data get more realistic, the discriminator network evaluation will have to be more robust. After the convergence, the discriminator should no longer be able to distinguish between real and fake data, and the generated instance will be close to the real data.

This dispute between the networks is possible due to the training loss function based on the cross-entropy formula. For the discriminator network we want to distinguish between

Figure 4 – GAN diagram where, given an input noise, data is generated and passed to the discriminator to distinguish between real and fake data.

real and fake data, so on Equation 2.1 we have x as the input data, y is the variable which determines if the data is real or fake and $D(x)$ is the discriminator network evaluation. When the data is real, the discriminator output should be 1, for the opposite, when the data is fake, the output should be 0. When evaluating real data, it will consider only the equation first term, i.e. $H(D(x), 1) = -\log(D(x))$. When evaluating fake data, only the equation second term will be considered, i.e. $H(D(x), 0) = -\log(1 - D(x))$. With the correct discriminator output the loss function $H(D(x), y)$ should approximate to zero, i.e. $\log(1) = 0$. On the opposite, when the discriminator fails, the loss function should increase to the infinite, i.e., $\log(0) \approx \infty$. By minimizing this loss function, the discriminator should learn to distinguish real and generated data.

$$H(D(x), y) = -(y \log(D(x)) - (1 - y) \log(1 - D(x))) \quad (2.1)$$

Equation 2.1 is the loss function for the discriminator network, which tries to separate real and fake data. For the generator network, we want the discriminator to evaluate generated data as real, i.e. $D(G(x)) = 1$. Thus, the loss function for the generator is displayed in Equation 2.2:

$$H(D(G(x))) = -\log(D(G(x))) \quad (2.2)$$

When $D(G(x)) = 1$ this objective function will be zero. By minimizing this loss function, the generated data will be enhanced to fool the discriminator.

Even with this smart model where two networks improve each other, some problems can appear. The loss functions from the discriminator and generator are minimized together, and this dependency can cause the gradient vanishing problem. It happens when the discriminator quickly learns how to distinguish the generated and real data. With this fast convergence on the discriminator network, the loss function quickly approximates to zero, diminishing the function gradient. With the gradient close to zero, the generator network learning got stagnated, and can not improve the data generation, achieving a poor result.

2.3.1 Wasserstein GAN

Depending on the data complexity which we are working with, the vanishing gradient may preclude the network learning, making impossible for the generator to create data close to real and limiting the use and application of GANs.

Arjovsky et al. (Arjovsky; Chintala; Bottou, 2017) propose an improved model to avoid the vanishing gradient problem, named Wasserstein GAN (WGAN). The idea is to replace the cross-entropy loss function by a distance function based on the Wasserstein Distance (OLKIN; PUKELSHEIM, 1982). This distance metric evaluates how far are two data distributions, by minimizing it, the network should be able to generate more realistic data. With this distance function, the discriminator will not classify the data as real or fake but define a value that describes the distance between both data distributions. That is why the evaluation network is named critic instead of discriminator. The Wasserstein distance guarantees a smoother gradient (Figure 5) on the loss function, which decreases the occurrence of the vanishing gradient problem.

Figure 5 – GAN and WGAN loss function gradient comparison. In blue, the real data and in green the fake data distribution. In red is displayed the cross-entropy function gradient, dividing real and fake data, where the vanishing gradient is highlighted. In light blue, the WGAN gradient is displayed, which is smoother every.

Source: Arjovsky, Chintala and Bottou (2017).

Equation 2.3 formulate the Wasserstein loss function, where P_r and P_g are the real and generated data distributions respectively, and $\Pi(P_r, P_g)$ denotes to the set of all joint distributions $\gamma(x, y)$ which the marginals are P_r and P_g . With this loss function, the network tries to define the distance between the two data distributions, giving a value that better describes the relationship

between them.

$$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} E_{(x,y) \sim \gamma} [\|x - y\|] \quad (2.3)$$

In the generator network, the loss function should approximate the generated data distribution to the real distribution. To do so, the loss function grows in the opposite direction of the critic network gradient (Equation 2.4). If the critic gives a high gradient value, the generator network will have a gradient with the same intensity, trying to diminish the distance to the real data.

$$G(Y) = -\nabla \frac{\sum_{i \in \{1, \dots, n\}} D(y_i)}{n} \quad (2.4)$$

2.3.2 Wasserstein GAN with Gradient Penalty

The WGAN guarantees a smoother gradient, which reduces the occurrence of the vanishing gradient problem. However, in some cases, this problem can still occur, to solve that a threshold clip is done when the gradient norm increases too much. This approach can solve the problem of gradient vanishing, however, due to different datasets aspects, the threshold value definition is not simple, and it is a rough solution with a rude interference on the learning process.

To limit the gradient norm and keep it stable a gradient penalty is proposed in (Gulrajani *et al.*, 2017). The idea is to keep the gradient norm at a central value between the range $(-1, 1)$. At each iteration, a penalty is calculated based on the gradient norm; that way, the rough clipping is replaced by a smooth and continuous penalty, maintaining the gradient norm centralized.

$$GP(G(x)) = \omega E_{G(x) \sim P_{G(x)}} (\|\nabla_{G(x)} D(G(x))\|_2 - 1)^2 \quad (2.5)$$

Equation 2.5 displays the gradient penalty formula. The discriminator gradient norm is computed, and, based on the distance to the boundaries, a penalty is calculated, which is multiplied by a defined parameter ω . The WGAN with Gradient Penalty (WGAN-GP) calculates the distance of the gradient norm to the boundary value $|1|$, and based on this, the gradient is penalized, stabilizing the gradient growth (Figure 6).

2.4 Final Considerations

In this chapter, we gave the theoretical background of our proposed methodology. We explained the differences between depth sensors and the main problems when dealing with stereo cameras. We introduced Generative Adversarial Networks, the concepts behind it, and how it is possible to generate data by allowing two networks to compete with each other. Also, the problem of the original GAN models is discussed, and the improved GAN models are introduced to solve

Figure 6 – WGAN gradient norm with different weight clipping and WGAN-GP. The different weight clipping changes the gradient norm drastically, behaving as a weight regulation, while the gradient penalty keeps the norm closer to a central value.

Source: [Gulrajani *et al.* \(2017\)](#).

this problem. Then, we define and discuss the WGAN and WGAN-GP models, introducing the main focus of our methodology.

The next chapter discusses the image processing tasks that can be used to solve our problem. We introduce and discuss works related to our proposal, and their positive and negative points are summarized.

LITERATURE REVIEW

This Chapter presents recent work on image processing focused on image reconstruction and object removal, on both RGB and depth images. Firstly we quickly define depth completion and inpainting tasks. Then, we discuss recent works in both areas, with their strengths and flaws when applied to our problem. In the end, we compare both methodologies and the joint application in our proposal for depth inpainting.

3.1 Image Inpainting

Image inpainting is the process of reconstruction in an image. With a specified marked area, the inpainting algorithm estimates the masked region based on the neighborhood information, achieving a realistic reconstruction. These methods can be applied to enhance noised image quality and to remove objects from the scene. In theory, image inpainting could deal with our problem of reconstructing disparity images; however, the difference between RGB and disparity data interferes with the disparity reconstruction precision.

3.1.1 *Related Works*

One of the most traditional image inpainting method was proposed in (CRIMINISI; PEREZ; TOYAMA, 2004) (Figure 7). This method firstly defines the boundary between the original and masked regions; then, an iterative algorithm analyzes the gradient of small patches around this borderline. After this analysis, the original structure is copied to the area below the boundary in a local evaluation. This algorithm runs until the whole masked area is filled. Arias et al. (ARIAS *et al.*, 2011) propose an improvement of this algorithm with a global image evaluation. The algorithm analysis grows in a larger region, helping the method to fill the patches with characteristics from farther areas.

Hervieu et al. (HERVIEU *et al.*, 2010) introduce a method to inpaint depth images.

Figure 7 – Inpainting process for object removal. (a) The original image; (b) the image with the woman removed from the scene.

Source: [Criminisi, Perez and Toyama \(2004\)](#).

The algorithm follows the idea from traditional inpainting methods by analyzing the gradient of the pixels. However, this method proposes an energy function to be minimized. The target region is filled by minimizing this objective function based on observable pixels, growing the inpainted area until the unknown area is estimated. A similar approach is proposed in ([CIOTTA; ANDROUTSOS, 2016](#)), however in that work, the estimated areas are matched with observable patches, and the characteristics of the real pixels are propagated to the generated area.

Those methods are based on real characteristics propagation, achieving notable results. Still, since they do an iterative region fill, for large removed areas, it is necessary a considerable time to process and reconstruct those images. Some recent works focus on deep networks to deal with the inpainting task, which processes images faster.

Iizuka et al. ([IIZUKA; SIMO-SERRA; ISHIKAWA, 2017](#)) proposes an encoder-decoder architecture to estimate a region of interest inside an image. The discriminator is divided in two, a local and a global discriminator. The local discriminator analyzes only the image generated area, while the global discriminator analyzes the whole image. This approach gives the network a guideline for a local consistency in the estimated pixels, and coherence between the generated region and real image. However, results show that for areas with more complex structures, the network fails in replicating these characteristics. To solve that a post-processing network is proposed in ([YU *et al.*, 2018](#)). This paper proposes a second network which processes the output from the encoder-decoder architecture. The post-processing network uses the same encoder-decoder architecture with the addition of a Contextual Attention branch, which matches and propagate real image characteristics to the generated area. Also, to have a more consistent convergence, the GAN loss function is replaced by a WGAN-GP loss function. This post-processing network increases the details on the generated pixels, replicating more realistic

texture characteristics.

Summarizing, inpainting algorithms achieve results close to real data based on an objective function minimization or a gradient analysis. The processing time problem could be solved by using GANs, becoming able to inpaint the masked area in real-time. The problem is that, methods able to remove large objects are mainly focused on RGB images, which are not readily applicable to depth maps. Methods focused on disparity data can not process the image in real-time, and can only reconstruct small regions in the image.

3.2 Depth Completion

Depth completion tasks are mostly focused on estimate disparity in shadowed areas. The lack of features in some regions on stereo images generate dark areas without a disparity value, due to that the final point cloud has some gaps without depth information. Depth completion focus on estimating those shadowed areas by doing a more thorough analysis of the stereo images and disparity neighborhood, to achieve an estimation for those regions. This task is also related to our problem; however, they are most focused on dark areas generated by the lack of features on the stereo matching step, which differs from our main problem.

3.2.1 Related Works

Most depth completion works use a Conditional Random-Field (CRF) method. CRF applied to image, models the pixels as a connected graph where each pixel is related to its neighbors. To estimate the pixel value, a function is defined where each value is conditional to the surrounding pixels. This function is optimized by a learning approach, trying to minimize or maximize the objective function. Other methods propose an energy function optimization based on RGB and depth data ([WEERASEKERA *et al.*, 2018](#); [ATAPOUR-ABARGHOU EI; GARANDERIE; BRECKON, 2016](#)). These approaches achieve relevant results since the depth is estimated based on the neighbor pixels information, ensuring consistency between them. Nevertheless, these approaches work well for completing small areas; for larger regions, the algorithms fails in estimating the center of the masked area since, on these regions, the energy function has only other estimated pixels as parameters. Without observable data to guide the reconstruction, the uncertainty of the estimated values increase.

A similar approach using Markov Random-Fields (MRF) is proposed in ([SHEN; CHEUNG, 2013](#)). This method models the image as a stochastic problem, where each pixel is related to the adjacent ones. Based on the assumption that a simple environment can be divided into background and foreground, this method tries to fill the holes in images captured in a controlled environment. A Bayesian Network is trained to differentiate the background from the foreground regions. Then, the objective function is divided into the data term and the smoothing term, regarding the depth estimation and a smoothing process, respectively. The problem of this work

relies on the focus on a well structured controlled environment, which can not be applied to complex scenarios.

Petrescu et al. (PETRESCU *et al.*, 2016) apply a temporal method relating a sequence of frames. This method uses the ray-marching algorithm, which starts from a non-observable pixel and marches all around by a parameterized angle step θ , until reaching an observable pixel. The angle and the distance from the start pixel to the observable pixel are used to weight the disparity value calculation. All the calculated disparity values for a pixel are used to compute the mean depth value, which will be the final estimation (Figure 8). The difference in this work is the temporal adaptation of this method. The ray-marching algorithm is applied, and previously estimated depth values are also considered. This gives a temporal coherence on the calculated disparity values, giving more information for the depth estimation. However, this method also fails in the estimation of large regions, since the ray-marching uncertainty increases in the center of large removed areas.

Figure 8 – Depth completion example. (a) Input depth maps where white regions are missing data; (b) the result from depth completion method; (c) difference between input and result image.

Source: Petrescu *et al.* (2016).

Figure 9 – Sparse Convolution results from Uhrig et al. (UHRIG *et al.*, 2017) compared with other methods.

Source: Uhrig *et al.* (2017).

Some approaches study and apply inpainting algorithms on disparity images to estimate depth occluded areas (ZHANG *et al.*, 2016). The inpainting method tries to estimate the pixel depth value; then, the result is smoothed based on the related RGB pixel value. This smoothing ensures a more consistent result since RGB images have more texture and structural information

to guide the pixel estimation. Buysens et al. (BUYSENS *et al.*, 2016) propose the use of morphological dilation operation to guide the inpainting gradient growth. Then, the inpainting process iteratively fills the occluded area using color information. Xue et al. (XUE; ZHANG; CAI, 2017) model the problem as a low-rank matrix, with this low-rank approximation, the missing values are estimated, and depth values filled. In images with small occluded areas to be estimated, the application of those inpainting methods can achieve a good result. However, those approximations uncertainty significantly increases when removing large areas, once the surrounding data already are depth estimations.

Uhrig et al. (UHRIG *et al.*, 2017) introduce the Sparse Convolution to deal with depth completion. LiDAR sensor data can be transformed into an image view, creating a depth map. However, since the LiDAR sensor capture fewer points than a stereo camera, the resultant image is too sparse. The idea of this network is to convolve the image with the called Sparse Convolution to do an interpolation of observable points and estimate the non-observable areas (Figure 9). Even with a dense, smooth result, this approach can not be applied to our task since this network uses many known points to estimate the unknown data. In our case, we try to estimate a largely unknown area, where, in many pixels, the estimation will not have observable data to guide the completion.

Some approaches extract semantic information from RGB to guide the disparity reconstruction. Zhang et al. (ZHANG; FUNKHOUSER, 2018) train a network to estimate the vectors normal to the surface of the pixels from RGB images. These data aggregate more information about the depth maps. Normal vectors can give the surface direction, which increases the knowledge about the depth structures. This new information is used to minimize an energy function, which considers the observable depth data, the surface normals, and adjacent pixels to reconstruct the unknown area. Auxiliary information is also extracted from RGB images in (ATAPOUR-ABARGHOUei; BRECKON, 2017). A network is used to do semantic label an image. This semantic information is used alongside depth data to complete the depth image based on many heuristics. The possible cases are listed, and each pixel to be estimated is classified as one of the twelve possibilities. Based on the case in which the pixel is classified, a heuristic is used to estimate the depth value. This method discretizes the possibilities of depth estimation and uses semantic information to help in the estimation task. In both cases, the methods propose the use of color images to complete the missing depth information. This completion is based on known RGB data to estimate the missing depth values. In our case, we have the information about one object, and we try to estimate what should be behind it if there was no object in that area. We base our estimation on observable surrounding depth information to reconstruct a large unknown masked area.

3.3 Final Considerations

In this chapter, we present recent works related to image inpainting and depth completion. We briefly described each approach and discussed their applications into our problem.

Most inpainting methods are focused on RGB images. Those methods have good visual results; still, they demand a high computational cost and have noisy outputs, which directly influence the point cloud generation when applied to disparity images. Some inpainting methods focus on depth images, but they can handle the problem only in small areas and in a well-controlled environment, which is impractical for our problem. The works that use GANs has the most recent state-of-art result; however, this solution is not readily applicable to our task due to the noisy aspect of the network output.

Depth completion works are mostly applied to complete small areas in a disparity image. The interpolation in these areas becomes easy since there are many observable pixels to guide the estimation. The works that can handle larger areas use known RGB information about the unknown depth area to estimate disparity information, completing the missing depth values.

Both tasks are close to our proposal. The inpainting methods achieve estimations without needing known information about the masked area. However, they take much time to reach the final result and, applied on depth data, the results are too noisy. Depth completion works achieve a smooth interpolation, estimating disparity data. However, they can handle only small regions and need previous knowledge about the area to be completed.

In the next chapter, we better discuss the problem and our methodology to deal with it. We present some specific aspects of depth data and the proposed solution to achieve a smooth disparity image inpainting.

PROPOSED WORK

This chapter presents our study about the main problem and our proposed network architecture to deal with disparity image reconstruction. We discuss disparity data sensitivity, and the impact of using specific depth features when processing it. We then apply state-of-art inpainting GAN on disparity images and highlight the flaws and differences when using RGB methods on depth maps. At last, we discuss our proposed approach and adaptations on the network architecture to focus on depth image reconstruction.

4.1 Inpainting GANs

To inpaint an image is necessary to estimate the target area based on the observable pixels. This reconstruction has to consider the main image characteristics, specifically the reconstructed area neighborhood. GANs are known for their ability to replicate a specific data domain; therefore, with a large dataset, those networks can reconstruct a masked region, maintaining the original image aspects. To train this network, a large set of images is necessary, and during the training, a random area in the image is erased and marked to be reconstructed. This random crop guarantees to the network a better reconstruction generalization since, at each step, it will have to use only the current reconstructed area neighborhood to guide the estimation. With the cropped and original images, we have our input data and the respective ground truth. After the training, the GAN will be able to reconstruct specified areas in an image, based only on its neighborhood without any information about what was in the masked region. Additionally, by masking a specific object, the network can reconstruct the area behind it using only the neighborhood data, i.e., the reconstruction will focus on the environment information, ignoring the fact that there was an object in that region. This approach is highly applicable to our problem since masking an object and reconstructing the environment behind it is the same as estimating the occluded information.

Iizuka et al. ([IIZUKA; SIMO-SERRA; ISHIKAWA, 2017](#)) formalized this approach and

proposed a GAN architecture able to reconstruct masked areas in an image. Their architecture consists of an encoder-decoder architecture with the generator and discriminator networks (Figure 10). They divided the discriminator network in two, a global and a local discriminator. The global discriminator evaluates the reconstructed area consistency within the whole image, analyzing both real and generated parts. The local discriminator evaluates only the generated image area, analyzing if this area would be sufficiently realistic by itself. The loss function for this network is the original GAN cross-entropy loss with an addition of a Mean-Squared Error (MSE) loss. With this architecture, they were able to achieve relevant results reconstructing RGB images of different domains. However, in some cases, the reconstruction was blurred and without sufficient texture information, which could be easily identified as generated data.

Figure 10 – Encoder-decoder inpainting GAN.

Source: Iizuka, Simo-Serra and Ishikawa (2017).

(YU *et al.*, 2018) proposed an improvement on this architecture by adding a post-processing network (Figure 11). This second network receives the coarse result from (IIZUKA; SIMO-SERRA; ISHIKAWA, 2017) network as input and, with a contextual attention branch, match and propagate real image characteristics to the generated area. Image pixels are normalized in the range $(-1, 1)$, to facilitate the network convergence. The WGAN-GP loss replaced the cross-entropy loss, and the MSE loss replaced by an L1 loss, which is the absolute difference between ground truth and generated images. To calculate the gradient penalty a mask m is used, where the removed pixels from the original image are equals to 1, and the original pixels equals to 0. This mask restricts the penalty calculation to the generated region (Equation 4.1).

$$GP(Y, m) = \omega E_{Y \sim P_Y} (\|\nabla_Y D(Y) \odot m\|_2 - 1)^2 \quad (4.1)$$

Furthermore, before calculating the L1 loss, a discounting mask m_s is applied to the generated image to weight pixels based on the nearest real neighbor. The use of WGAN-GP and L1 loss helps the network to improve its results. While the L1 loss tries to approximate the generated image to the ground truth, the WGAN-GP loss evaluates how real those results

Figure 11 – Generative Inpainting Network architecture. The coarse result is the output from the original encoder-decoder architecture, post-processed with the attention branch in the refinement network.

Source: Yu *et al.* (2018).

look. Both loss functions are applied to the final and coarse image results, to improve the feature extraction in all the network stages. With those adaptations, this architecture uses the coarse result from the first network to improve and propagate real characteristics to the generated pixels.

4.1.1 Color Inpainting in Disparity Data

The network proposed by (YU *et al.*, 2018) shows a coherent reconstruction in color, texture, and structural levels, quickly completing the masked area. This approach seems to be easily applicable to disparity images once a deep network can extract many features from the input data. Since we have a representative amount of training data, the network should be able to reconstruct depth maps. However, as we can see in Figure 12, the results are not as good as expected. There is noise in the reconstructed disparity area, which has significant influence in the generated point cloud. The deep network should be able to extract features from disparity images due to the number of convolutions and the kernel weights fitting. Even so, noisy disparity data is generated, which directly affects the tridimensional points.

4.2 Proposed Architecture

Disparity images encode depth information in one-channel color space. Therefore, the pixel error from the ground truth image is proportional to the error from generated and real depth values. The problem when dealing with depth data is that the neighbor depth values are related to each other. The difference between two neighbor pixels gives information about the object that the pixels rely on. If the difference between two pixels is too high, it means that there is a slope in that area, or the pixels are on distinct structures. An individual pixel color evaluation is not sufficient for coherent disparity structures reconstruction.

When analyzing the loss function from (YU *et al.*, 2018), it is possible to notice that the

Figure 12 – Contextual attention network applied to RGB and disparity images, evidencing the reconstructed area.

error is evaluated individually, i.e., the error in a pixel does not consider the adjacent values. This evaluation can increase the noise on neighborhood pixels transitions. This noise may not be noticed on color images once, for a visual purpose, this error is not so representative. However, for disparity images, the noise is evidenced once the point cloud is reconstructed, and this transition noise can be easily noticed. This analysis shows that the main problem is related to the loss function and the network result evaluation. The network can extract disparity features, although the proposed loss function is not sufficient to evaluate the depth map reconstruction.

4.2.1 Depth Surfaces

Surface features can be extracted from depth and disparity data (BESL; JAIN, 1986) (NAKAGAWA *et al.*, 2015) (BADINO *et al.*, 2011) (HARMS *et al.*, 2014). In (NAKAGAWA *et al.*, 2015), depth values are estimated based on the disparity data and camera parameters, then a vector normal to a pixel is calculated based on adjacent pixel values (Figure 13). Initially the depth image is smoothed with a bilateral filter to diminish noises. Then, the gradients ∂X , ∂Y , ∂Z are calculated as formulated in Eq. 4.3, where $X(x,y)$, $Y(x,y)$ and $Z(x,y)$ are the functions

that calculate the (X, Y, Z) point based on the pixel value (x, y) and the stereo camera parameters.

$$\begin{aligned}
 \frac{\partial X(x, y)}{\partial x} &= \frac{Z(x, y)}{f} + \frac{(x - c_x)}{f} \frac{\partial Z(x, y)}{\partial x} \\
 \frac{\partial Y(x, y)}{\partial x} &= \frac{(y - c_y)}{f} \frac{\partial Z(x, y)}{\partial x} \\
 \frac{\partial X(x, y)}{\partial y} &= \frac{(x - c_x)}{f} \frac{\partial Z(x, y)}{\partial y} \\
 \frac{\partial Y(x, y)}{\partial y} &= \frac{Z(x, y)}{f} + \frac{(y - c_y)}{f} \frac{\partial Z(x, y)}{\partial y}
 \end{aligned} \tag{4.2}$$

Where f is the focal distance from stereo cameras and (c_x, c_y) is the camera focal point. With the vectors $\vec{v}_x(\frac{\partial X(x, y)}{\partial x}, \frac{\partial Y(x, y)}{\partial x}, \frac{\partial Z(x, y)}{\partial x})$ and $\vec{v}_y(\frac{\partial X(x, y)}{\partial y}, \frac{\partial Y(x, y)}{\partial y}, \frac{\partial Z(x, y)}{\partial y})$ a normal vector is calculated as $\vec{n} = \vec{v}_x \times \vec{v}_y$. With this formulation the paper estimate a normal vector to each disparity pixel. The problem on using depth values is that this is not easily generalized, since each stereo camera has different parameters, and there is a pixel positioning dependency in the depth value calculation. When using different sized images on the network, this dependency of pixel positions could lead to an overfitting of the network, based on the image size.

Figure 13 – Relation between adjacent pixels and the surface normal vector calculation.

Source: Nakagawa *et al.* (2015).

In our work, we try to use a generalized solution for disparity images, which means that the camera parameters would not be necessary. Based on the formulation proposed in

Figure 14 – Normal vectors estimation from color variation around x and y image axis. In green, the y parallel vector, in red, the x parallel vector and in blue the vector normal to the two vectors surface.

(NAKAGAWA *et al.*, 2015) and using the definition of normal vectors, we can generalize a normal vector calculation for each disparity pixel based on the neighborhood values. Given two orthogonal vectors $\vec{v}$ and $\vec{n}$, the cross product between them $\vec{v} \times \vec{n}$ gives a resultant vector normal to surface which $\vec{v}$ and $\vec{n}$ belongs.

The image directional gradients give the color change around a pixel in a given direction. To compute the gradients of a pixel $P_{i,j}$, we calculate the differences in X and Y image axis. This gradient provides information about the color variation around one pixel. Since the disparity pixel color encodes the depth values, the mean variation on the disparity image axis $P_{\Delta i}$ and $P_{\Delta j}$ is proportional to the depth variation around pixel $P_{i,j}$. Therefore, to calculate the vector normal to a pixel, we take two unitary orthogonal vectors parallel to the two image axis. Then, we define their Z component as the respective directional gradient value on that pixel. Thus, we have two orthogonal vectors pointing to the disparity variation in their respective axis and, consequently, parallel to the pixel surface. The cross product between them gives a resultant vector normal to the surface, which the pixel $P_{i,j}$ belongs (Figure 14). In the end, the estimated normal vector is normalized (Eq. 4.3), and the unitary vector normal to the pixel surface is defined:

$$\begin{aligned}
 P_{\Delta i} &= \frac{P_{i+1,j} - P_{i-1,j}}{2} \\
 P_{\Delta j} &= \frac{P_{i,j+1} - P_{i,j-1}}{2} \\
 \vec{v}_{\Delta i} &= (1.0, 0.0, P_{\Delta i}) \\
 \vec{v}_{\Delta j} &= (0.0, 1.0, P_{\Delta j}) \\
 \vec{v} &= \vec{v}_{\Delta i} \times \vec{v}_{\Delta j} \\
 \vec{n} &= \frac{\vec{v}}{\|\vec{v}\|}
 \end{aligned} \tag{4.3}$$

As we can see in Figure 15, the generalized formulation for disparity images can represent the direction of the different structures, and the distinct surfaces can be well distinguished. Even without the real depth value calculated with the stereo camera parameters, we could estimate

normal vectors based on the disparity color variation.

Figure 15 – Surface normals image generated by the generalized formulation from the disparity image.

4.2.2 Vectorial Loss

With the surface normals estimation, we have a vectorial representation, that is calculated based not in an individual value, but a neighborhood analysis. This visualization ensures a more global evaluation, considering adjacent pixels, and its influence in the estimated disparity value. We maintain the loss function proposed in the Contextual Attention network and add a new factor to be minimized that we call Vectorial Loss. Since the surface normals can be estimated by the four neighbor pixels without the stereo parameters, we can easily model this calculation as a convolution (Figure 16), and add it to the GAN training without significantly increase the processing time. At each training step, we estimate the normals from the ground truth and generated disparity images. We also use the L1 metric, calculating the absolute difference between each axis (x, y, z) in the normal vector $\vec{n}$, and multiply by a fixed parameter α . The discriminator loss d_l and generator loss g_l are defined as:

Figure 16 – Kernel for surface normal estimation.

$$\begin{aligned}
 d_l &= W(P_r, P_g) + GP(Y, m) \\
 g_l &= \beta G(Y) + \phi L1(X, Y) + \alpha V_l(X_{\vec{n}}, Y_{\vec{n}})
 \end{aligned}
 \tag{4.4}$$

Where V_l is our Vectorial Loss function, and $X_{\vec{n}}$ and $Y_{\vec{n}}$ are the surface normals from ground truth and generated disparity images.

This new loss factor (Figure 17), give to the network a better guideline for depth estimation by analyzing the generated structures surfaces. Since normals are calculated by adjacent pixels, this will decrease the individual evaluation aspect, and increase the neighborhood analysis.

Figure 17 – Contextual Attention network diagram with surface normals estimation and Vectorial Loss.

4.2.3 Surface Attention Branch

The Contextual Attention branch's main idea is to compare patches of pixels to propagate real image characteristics to the generated area. This branch divides the image into two classes, foreground as the generated area, and background as the real image. Then, the background area is divided into 3×3 patches. The background patches are reshaped as convolutional filters to guide the characteristics propagation. The cosine similarity is used to calculate the similarity between foreground $f_{x,y}$ pixels and background $b_{x',y'}$ normalized patches. The similarity $s_{x,y,x',y'}$ is quantified for each pair of foreground pixel and background patches by:

Figure 18 – Contextual Attention branch diagram.

Source: Yu *et al.* (2018).

$$s_{x,y,x',y'} = \left\langle \frac{f_{x,y}}{\|f_{x,y}\|}, \frac{b_{x,y}}{\|b_{x,y}\|} \right\rangle \quad (4.5)$$

Also, the similarity scores are weighted with softmax in x', y' axis:

$$s_{x,y,x',y'}^* = \text{softmax}_{x',y'}(\lambda s_{x,y,x',y'}) \quad (4.6)$$

Where λ is a fixed parameter.

After that, closer patches are encouraged to propagate the same neighborhood characteristics. This neighborhood propagation is done by a left-right and top-down attention score shift, in a kernel of k size. This propagation is done by the sum of the kernel attention scores (Eq. 4.7). By summing the attention scores, the pixel inherits the neighborhood closeness to the matched real patches.

$$s_{x,y,x',y'}^+ = \sum_{i \in \{-k, \dots, k\}} s_{x,y,x',y'}^* \quad (4.7)$$

This approach propagates the characteristics in the surrounding patches and can easily be implemented as a convolution, which does not highly increase the processing time.

With the real patches and generated pixels matched, the background reshaped convolutions are used to deconvolve the matched pixels, propagating real characteristics to the generated area. However, when dealing with disparity images, the propagation of disparity values only is not so relevant, once the disparity pixel value is not related to the object context that it relies on. Disparity values can not well represent the relationship between surrounding pixels, that is why it is necessary the analysis of neighbor depth values for a more coherent characteristics propagation. If we propagate the characteristics based only on disparity pixels, we may propagate depth values to neighborhood areas where it does not correctly fit, generating irregularities in structures surfaces.

By studying the Contextual Attention branch application in disparity images, it is possible to notice that the patches' characteristics propagation can add noise to the generated area. This happens because the depth values do not carry the object context, but only the distance of the point to the camera. To solve that, we add to the Attention branch also the surface normal vectors alongside the disparity image. This information aggregate to the branch the context of the surface in which the pixel belongs. Also, since normals calculation uses the neighbor pixels to estimate the vector, in the transition between patches from the same surface, the attention score tends to be closer to each other, reinforcing the adjacent patches' dependency. The attention score will be calculated by the depth values and structures surface similarities, and the real patches shaped convolution will propagate both, real surface characteristics and disparity values.

The Surface Attention branch (Figure 19) increases the similarity score between adjacent surface patches since the estimation of the normal vector is based on a kernel formulation. This increase of similarities improves the reconstruction smoothness on the transition between patches and different structures surfaces, achieving a depth distribution closer to the real image. By

considering the surface context in the attention score calculation, the irregularities generated by the Contextual Attention branch can be reduced.

Figure 19 – Contextual Attention network diagram with surface normals estimation, Vectorial Loss, and Surface Attention adaptation.

4.2.4 Surface Discrimination

The GANs architectures are based on two networks where one generates data based on an input dataset, and the other evaluate how real this generated instance looks. Following this idea, our network should generate the output depth map, and the discriminator should evaluate how real this disparity reconstructed image looks. In our architecture, we propose to evaluate both the disparity image and respective surfaces normals by adding the Vectorial Loss factor to be minimized. However, adding only this loss function to guide the surface reconstruction ignores the most important advantage of using GANs, the discriminator's capability of analyzing real and generated data distributions. Therefore, to enhance the generated surface evaluation, we give to the discriminator network the disparity images and the respective surface normals. This approach guarantees that the discriminator will evaluate the target disparity output image, but will also evaluate if the correspondent surfaces look real. That way, as the network tries to improve and minimize the disparity image error, it will also ensure that one improvement on disparity data will not spoil the generated surfaces, ensuring the coherence on the disparity values and their respective surfaces.

Figure 20 displays our final proposed architecture, where we look after depth features evaluation at many network stages. Those adaptations focus the network generalization on depth data, to achieve a more realistic disparity reconstruction.

4.3 Final Considerations

In this chapter, we revise the Contextual Attention network and present our proposed loss function. Based on our study about depth features, we propose a reformulation of the loss function, aggregating an evaluation of the surface normal vectors from generated disparity images. Also, we add to the contextual attention branch and the discriminator network the surface

Figure 20 – Our proposed GAN architecture with surface normals analysis in many of the data processing stages.

normal vectors, to propagate real surface characteristics, and evaluate how real the generated surfaces look. The idea of using depth features to evaluate the result is to give to the network a better data distribution visualization, helping the convergence of convolutions weights towards the ground truth surfaces in depth data distribution.

In the next chapter, we present our results using pixel-wise error metrics, and we discuss the impact of such metrics on GANs performance evaluation and how they can mislead the network analysis. Then, we discuss different data distribution metrics and use them to incrementally evaluate each adaptation described in the subsections of this chapter, to see how each modification could improve the network results. In the end, we compare our proposed architecture with the original state-of-art inpainting networks, incrementally applying our proposed adaptations.

RESULTS

In this chapter, we present our proposed loss function and architecture results compared with state-of-art networks. First, we introduce the disparity image datasets used for training. We then present the qualitative and quantitative results with MSE and Vectorial Error (VE) metrics and discuss this preliminary evaluation. Furthermore, we discuss the GAN generated data aspects and how pixel-wise metrics can mislead the network evaluation. Then, we argue about data distribution comparison metrics and their applicability to GANs evaluation. At last, we present the quantitative analysis using data distribution metrics to precisely evaluate the difference between generated and real data distributions.

5.1 Training Datasets

To effectively test our network, it is necessary stereo images from the real world in an outdoor urban environment. This data represents the scenario in which an autonomous vehicle would be inserted. The KITTI dataset ([GEIGER *et al.*, 2013](#); [GEIGER; LENZ; URTASUN, 2012](#)) is a benchmark dataset on the ITS field. This dataset capture many images in different weather conditions around the streets of Karlsruhe, Germany (Figure 21). To calculate the disparity values from stereo images we have used the Efficient Large-Scale Stereo Matching (ELAS) ([GEIGER; ROSER; URTASUN, 2011](#)) algorithm. However, when calculating disparity data, the lack of image features can generate shadowed gaps without disparity information on the resultant depth map. Due to that, we also have used the CityScapes ([CORDTS *et al.*, 2016b](#)) dataset. This dataset collects images from around 50 cities in Germany, which covers a considerably larger area. The images are labeled for semantic and instance segmentation, and for each image, there is a related depth map (Figure 22). The images are manually selected, so they have less noise than the KITTI dataset calculated disparity images. With this denser disparity dataset, we can diminish the noise influence during the network training, improving the environment structures reconstruction.

Since on both datasets the images are larger than the training images, we can divide

each dataset image into sub-images, increasing the amount of data. After cropping the original images, we defined a training set with 11.118 images and a validation set with 2.312 images for the CityScapes dataset. For the KITTI dataset, a smaller set with 1.813 training images and 183 images for validation were defined.

Figure 21 – Different scenes examples captured in the KITTI Dataset.

Source: [Geiger et al. \(2013\)](#).

Figure 22 – CityScapes examples with the RGB image and depth maps partially overlaid, CityScapes instance labelling and different semantic segmentation approaches using their dataset.

Source: [Cordts et al. \(2016b\)](#).

To train the networks, we follow the training steps used on the ([YU et al., 2018](#)) network. Squared images of 256×256 size are used for training with a random 128×128 cropped area

masked to be generated by the network. It is necessary to highlight that, for this application, the ground truth is somewhat complex to define. During the training, the original images without cropping are the ground truth, which is what we want the network to learn to reconstruct. However, after the training, we can apply this network on different tasks - e.g., object removal, depth completion - and in that case, we may not have a ground truth to evaluate the reconstruction, leaving only the qualitative analysis.

To evaluate our methodology, we have used qualitative and quantitative analysis. For the quantitative evaluation, we use the validation images also with a random 128×128 crop around it. The network then reconstructs the cropped region, and, with the original image as ground truth, we compute the reconstructed image error. That way, we can numerically compare the original network with our proposed architecture to evaluate our methodology. For the qualitative evaluation, we apply the network on different images, marking different objects for removal. With this analysis, we can examine the network depth reconstruction and compare our results with other inpainting networks.

5.2 Results

First we test our Vectorial Loss on two networks, the encoder-decoder architecture from (IIZUKA; SIMO-SERRA; ISHIKAWA, 2017) and the improved network proposed in (YU *et al.*, 2018). We compare the network results with and without the Vectorial Loss, to evaluate the influence of our proposed loss function on depth reconstruction. Second, we evaluate our proposed architecture and compare it with the state-of-art Contextual Attention network. We incrementally apply our adaptations to the Contextual Attention architecture, retrain the network and calculate their error. With this incremental evaluation, we can verify how each modification influences the network disparity reconstruction. For all the networks training, we have used the same parameters on the generator loss function:

$$g_l = \beta G(Y) + \phi L1(X, Y) + \alpha V(X_{\vec{n}}, Y_{\vec{n}}) \quad (5.1)$$

Where $G(Y)$ is the GAN generator loss, the $L1(X, Y)$ is the L1 metric between the generated data and the ground truth images, and $V(X_{\vec{n}}, Y_{\vec{n}})$ is the Vectorial Loss applied on the surface normal vectors from generated and ground truth images. The defined parameters for each loss function were $\beta = 0.0005$, $\phi = 1.2$ and $\alpha = 0.8$.

5.2.1 Vectorial Loss

In this first test, we want to quantify only how our loss function can improve the network depth reconstruction. Since this modification only quantifies the pixel-wise error between the surfaces, the shadowed regions will not have a significant interference on the training. Due to that,

Table 1 – Encoder-decoder and the Contextual Attention networks with and without the Vectorial Loss

	MSE	VE
E.D.	7.05%	22.67%
E.D. + VLOSS	6.79%	17.34%
C.A.	8.08%	19.53%
C.A. + VLOSS	7.41%	14.74%

we use the KITTI dataset, which has a smaller set of images and is faster to train. Table 1 displays the MSE and VE errors. On both networks, our loss function could improve the results on depth and surface data. By quantifying the surface error, the network learns to generate smoother surfaces, implying on a more homogeneous disparity distribution. The Vectorial Loss guides the network data generation, highlighting the surfaces and disparity neighborhood distribution importance on the depth estimation, improving the environment reconstruction.

5.2.2 Proposed Architecture

To test our proposed depth inpainting GAN architecture, we have used the same quantitative evaluation. We incrementally apply our proposed modifications and compare the results. Since on these tests we evaluate the characteristics propagation and surface discrimination, a noisy depth map may spoil the reconstruction, once the shadowed areas can also be matched and propagated. Hence, we have used the CityScapes dataset, which has a larger and denser set of images.

Figure 23 displays each network depth reconstruction. The original network can reconstruct the disparity image; however, some artifacts are generated around the estimated region. Besides that, the reconstructed surfaces have a higher color variation when compared with the ground truth. This color variation means that the vectors normal to the surfaces of the pixels also have high variance, representing uneven surfaces. When we apply the Vectorial Loss, the disparity interpolation is smoother, yet, this smoothness interferes with the generated surfaces, which deviates from the original surface data distribution. With the Surface Attention branch, the network tries to replicate more realistic surface characteristics, proposing more complex structures. However, since these proposed surfaces are not discriminated, they may do not fits in the region where they are placed, generating irregular structures. When we add the Surface Discriminator, the network creates complex and coherent structures, achieving a more consistent reconstruction. Also, the network understands that too smooth surfaces do not look realistic. Due to that, the surfaces created by the network inherit the real surface roughness, approximating the generated data distribution to the ground truth data.

Nevertheless, when we calculate the MSE and VE for those networks, the quantitative result does not point to those improvements (Table 2). This behavior can be explained by the

Table 2 – Mean-Squared Error and Vectorial Error calculated from each network adaptation and original Contextual Attention network.

	MSE	VE
CA (YU <i>et al.</i> , 2018)	189.078	0.1212
CA (YU <i>et al.</i> , 2018) + VL	181.870	0.0908
SA + VL	194.335	0.0907
Ours	197.527	0.1054

way that GANs works.

Figure 23 – Networks disparity and surface reconstruction comparison. (a) Original RGB, disparity and surface images with the cars to be removed; (b) original network; (c) network with Vectorial Loss; (d) network with Vectorial Loss and Surface attention branch; (e) our proposed architecture with Vectorial Loss, Surface Attention branch and Surface Discriminator.

When we add only the Vectorial Loss, the network tries to minimize the error for each pixel locally, i.e., a mean disparity value for a neighborhood is calculated, and the neighbor pixels fluctuate around this value to diminish the error. As it is possible to notice in Figure 23.c, the network with the Vectorial Loss proposes a smoothness around all the generated area. Our final proposed architecture understands that this smoothness does not look realistic, and proposes

an interpolation with a subtle roughness as in the real data. However, when evaluating the image with pixel-wise metrics, the error can increase since the pixel where those roughness are placed may not match with the ground truth (Figure 24). Therefore, pixel-wise evaluation metrics can mislead the performance analysis of the network.

Figure 24 – Mean squared pixel error from the network with Vectorial Loss only and with Vectorial Loss and Surface Discrimination.

5.2.3 Statistical Distance Metrics

Statistical distance metrics can quantify the difference between two data distributions. (Arjovsky; Chintala; Bottou, 2017) propose the WGAN network with a loss function based on the Wasserstein distance metric. Some other distribution distance metrics are discussed on that work, e.g., Kullback-Liebler and Jensen-Shannon. Since we are dealing with images, we can represent the data distribution as histograms, and use those distance metrics to compare the reconstructed and ground truth images. In addition to these, other metrics can also be used to compare histograms. The histogram intersection is a metric often used for image comparison; this method calculates the percentage of matching bins between two image histograms (SWAIN; BALLARD, 1992). Also, the correlation between histograms can be calculated, quantifying how much both images are related. By using those different histograms and data distribution distance metrics, we can compare real and generated data distributions, withdrawing the local evaluation from pixel-wise metrics. Thus, we can analyze how well the network can replicate real disparity data.

For our quantitative evaluation, we have used the Wasserstein distance, Kullback-Liebler and Jensen-Shannon divergences, Histogram Intersection, and Histogram Correlation metrics. The Wasserstein, Kullback-Liebler, and Jensen-Shanon are distance metrics, i.e., higher values represent farther data distributions. Histogram Intersection and Histogram Correlation give a similarity percentage between both data distributions, i.e., values closer to one imply that both data distributions are strongly related. We apply those metrics on the disparity image and the surface normal vectors to evaluate both depth and surface data distributions.

Table 3 – Data distribution distances between ground truth and reconstructed images for each applied adaptation. CA is the original Contextual Attention network (YU *et al.*, 2018); VL is the same network with our Vectorial Loss function; SA the network with Vectorial Loss and Surface Attention branch adaptation; Ours is our final proposed network with Vectorial Loss, Surface Attention branch and Surface Discrimination.

	Jensen-Shannon		Kullback-Liebler		Wasserstein		Hist. Intersection		Hist. Correlation	
	Depth	Surface	Depth	Surface	Depth	Surface	Depth	Surface	Depth	Surface
CA	0.4635	0.3845	0.6979	0.6536	0.0025	0.0021	0.5085	0.5667	0.6135	0.4258
VL	0.4749	0.3887	0.7428	0.6124	0.0024	0.0015	0.4917	0.5520	0.5839	0.4727
SA	0.4550	0.3862	0.6681	0.6066	0.0026	0.0015	0.5206	0.5557	0.6620	0.4772
Ours	0.4556	0.3797	0.6726	0.5987	0.0025	0.0018	0.5232	0.5660	0.6583	0.5126

Table 3 displays the data distributions distances on all four trained network architectures. Those results follow the qualitative evaluation, as we add our adaptations, the generated image data distribution gets closer to the real data. When we add the Vectorial Loss, the network disparity data get away from the ground truth; however, we can notice an improvement in the surface data distribution. When we apply the Surface Attention branch, the network can replicate surface and disparity characteristics, due to that, both depth and surface data distribution are improved. Last, when evaluating our final proposed architecture with Surface Discriminator, we notice a slight disparity error increase on some metrics at the expense of a significant improvement on the surface distribution. Thus, even the network proposing structures with some depth error, their surfaces will be more realistic. Those results agree to our network qualitative evaluation, where our generated images look more realistic than state-of-art results.

When comparing both pixel-wise and statistical distance metrics we notice that, on some applications, pixel-wise metrics can mislead the network performance evaluation. In our case, where disparity data are too sensible, those metrics could not correctly portray our network results. With statistical distance metrics, we were able to understand how each modification influenced on the network data generation. This analysis evidence the relevance of data distribution metrics on GANs evaluation. Once we deal with data distribution imitation, data distribution comparison metrics are very suitable to evaluate GANs performances.

Furthermore, for a more embracing qualitative evaluation, Figures 25 and 26 displays many different objects removed from different disparity images compared with the Contextual Attention network, and Figure 27 displays only our proposed network result applied to more object removal examples on different scenes. On those images, we can compare the original and our final proposed network architecture applied to depth inpainting and see how our results exceed the state-of-art Contextual Attention network, strengthening our hypothesis validation.

For a more complete analysis of our results, in this [video](#)¹ we remove objects from a sequence of frames from CityScapes demo dataset.

¹ Video URL: <https://www.youtube.com/watch?v=0fQ3vPD88-w>

Figure 25 – Comparison between Contextual Attention network (YU *et al.*, 2018) depth reconstruction and ours. First column the RGB image; Second column the original disparity image; Third column the Contextual Attention reconstruction of the area behind the target objects; Last column our reconstruction.

5.3 Final Considerations

In this chapter, we present and discuss our results. By applying our Vectorial Loss in two different networks, we highlight the relevance of our depth focused loss function, since in both cases, a significant improvement was achieved.

When evaluating our proposed depth inpainting architecture using pixel-wise metrics, the quantitative results diverge from the qualitative evaluation. However, when applying data distribution distance metrics, we notice that pixel-wise metrics may not correctly portray the network results. Distribution distance metrics could better portray our GAN results.

Finally, when comparing the disparity reconstructed images, our network could surpass the original state-of-art inpainting architecture. Our adapted network achieved a coherent disparity and surface reconstruction, generating a consistent and realistic depth interpolation.

Figure 26 – Comparison between Contextual Attention network (YU *et al.*, 2018) depth reconstruction and ours. First column the RGB image; Second column the original disparity image; Third column the Contextual Attention reconstruction of the area behind the target objects; Last column our reconstruction.

Figure 27 – Examples of object removal on different scenes with our proposed network.

CONCLUSIONS

Recent studies on stereo matching algorithms increase the precision and reliability of image-based depth sensors. Dense point clouds and color information captured by stereo cameras ensure more dense environment information when compared with LiDAR sensors. Once autonomous vehicles are each day closer to be applied in a real city environment, where there are dynamic obstacles and constant changes happen, this data is essential for the vehicle to have as many environment information as possible for a reliable decision making.

When compared with humans drivers, intelligent vehicles lacks on the inference of environment occluded information. Human drivers can deduce what is behind objects and structures based on previous knowledge, making decisions based on facts and assumptions about the environment. In perception tasks in which obstacles are not of interest, occlusion generated by these objects interferes on the perception system once it can not guess what is behind the object. Consistent estimation of the occluded area could help the perception system to understand the occluded environment and make a coherent decision. In the literature review, we could find image processing methods able to estimate a region of interest based on neighbor pixels, which, in theory, could be applied to deal with occlusion. However, it was found a clear gap when dealing with disparity images. The discussed methods deal with color images, although, when applied to depth maps, the noise became a problem when generating the point cloud from the reconstructed image.

We propose a GAN focused on depth features analysis able to inpaint disparity images, reconstructing depth environment information. Our results highlight the importance of loss functions specialized in the target training data. The Vectorial Loss helped the network to generate smoother depth interpolations, decreasing the generated surface error. Also, our proposed architecture analyzes the generated disparity and surface data, improving its consistency. On our final results, we could surpass state-of-art inpainting networks when applied to depth inpainting. Also, regarding GANs evaluation, we expose the pixel-wise metrics problems. Since we deal with data distribution imitation, statistical distance metrics are better suited to portray GANs

performance. Our main contribution is a loss function focused on depth features along with a GAN able to inpaint disparity images and the proposal of statistical analysis for evaluating GANs.

In future works, this network could be used to estimate occluded depth information. With appropriate adaptations, it could help improving object tracking methods during occlusions. By estimating the occluded object position and knowing its shape; with this network, it would be possible to recreate the occluded object and the environment surrounding it, maintaining the tracking and understanding the object context even during the occlusion. Another application could be estimating objects occlusions during environment mapping. Removing the dynamic objects in the scene and estimating the information hidden by those objects would increase the amount of data collected for the map construction, creating a more complete and dense map. Many other possible applications for this work may arise, with a network able to estimate and helping understand information that can not be collected by the sensors can help perception systems to have a more complete environment knowledge, increasing their accuracy. There are yet many other possible applications, we expect that this work may help the ITS field to achieve notable advances, that is why we make our code available in this [repository](#).

During this work, we could contribute with a paper that was published at the Hawaii International Conference on System Sciences 2019 ([BRUNO *et al.*, 2019](#)). Also, as a consequence of the relevance of our results, we could publish it at the 30th IEEE Intelligent Vehicles Symposium (IV'19) held in Paris ([Matias *et al.*, 2019](#)), which were also selected for oral presentation. The final results of our work were published as a preprint on arXiv ([MATIAS; SOUZA; WOLF, 2019](#)), where we discuss the final version of our network and with more consistent analysis of our results.

BIBLIOGRAPHY

Aloysius, N.; Geetha, M. A review on deep convolutional neural networks. In: **2017 International Conference on Communication and Signal Processing (ICCSP)**. [S.l.: s.n.], 2017. p. 0588–0592. ISSN null. Citation on page 27.

ARIAS, P.; FACCIOLO, G.; CASELLES, V.; SAPIRO, G. A variational framework for exemplar-based image inpainting. **International Journal of Computer Vision**, v. 93, n. 3, p. 319–347, Jul 2011. ISSN 1573-1405. Available: <<https://doi.org/10.1007/s11263-010-0418-7>>. Citation on page 33.

Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. **arXiv e-prints**, p. arXiv:1701.07875, Jan. 2017. Citations on pages 29 and 56.

ATAPOUR-ABARGHOUEI, A.; BRECKON, T. Depthcomp : real-time depth image completion based on prior semantic scene segmentation. In: **28th British Machine Vision Conference (BMVC) 2017**. British Machine Vision Association (BMVA), 2017. Available: <<http://dro.dur.ac.uk/22375/>>. Citation on page 37.

ATAPOUR-ABARGHOUEI, A.; GARANDERIE, G. P. de L.; BRECKON, T. P. Back to but-terworth - a fourier basis for 3d surface relief hole filling within rgb-d imagery. In: **2016 23rd International Conference on Pattern Recognition (ICPR)**. [S.l.: s.n.], 2016. p. 2813–2818. Citation on page 35.

BADINO, H.; HUBER, D.; PARK, Y.; KANADE, T. Fast and accurate computation of surface normals from range images. In: . [S.l.: s.n.], 2011. p. 3084 – 3091. Citation on page 42.

BADRINARAYANAN, V.; HANDA, A.; CIPOLLA, R. Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. **arXiv preprint arXiv:1505.07293**, 2015. Citation on page 22.

BECATTINI, F.; BERLINCIONI, L.; GALTERI, L.; SEIDENARI, L.; BIMBO, A. D. Semantic road layout understanding by generative adversarial inpainting. **CoRR**, abs/1805.11746, 2018. Available: <<http://arxiv.org/abs/1805.11746>>. Citation on page 23.

BESL, P. J.; JAIN, R. C. Invariant surface characteristics for 3d object recognition in range images. **Computer Vision, Graphics, and Image Processing**, v. 33, n. 1, p. 33 – 80, 1986. ISSN 0734-189X. Available: <<http://www.sciencedirect.com/science/article/pii/0734189X86902203>>. Citation on page 42.

BRUNO, D. R.; MATIAS, L. P. N.; AMARO, J.; OSÓRIO, F. S.; WOLF, D. F. Computer vision system with 2d and 3d data fusion for detection of possible auxiliaries routes in stretches of interdicted roads. In: **HICSS**. [S.l.: s.n.], 2019. Citation on page 62.

BUYSENS, P.; MEUR, O. L.; DAISY, M.; TSCHUMPERLÉ, D.; LÉZORAY, O. Depth-guided disocclusion inpainting of synthesized rgb-d images. **IEEE Transactions on Image Processing**, v. 26, p. 525–538, 2016. Citation on page 37.

CIOTTA, M.; ANDROUTSOS, D. Depth guided image completion for structure and texture synthesis. In: **2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.: s.n.], 2016. p. 1199–1203. ISSN 2379-190X. Citation on page 34.

CORDTS, M.; OMRAN, M.; RAMOS, S.; REHFELD, T.; ENZWEILER, M.; BENENSON, R.; FRANKE, U.; ROTH, S.; SCHIELE, B. The cityscapes dataset for semantic urban scene understanding. **CoRR**, abs/1604.01685, 2016. Available: <http://arxiv.org/abs/1604.01685>. Citations on pages 22 and 23.

_____. The cityscapes dataset for semantic urban scene understanding. In: **The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2016. Citations on pages 51 and 52.

CRIMINISI, A.; PEREZ, P.; TOYAMA, K. Region filling and object removal by exemplar-based image inpainting. **IEEE Transactions on Image Processing**, v. 13, n. 9, p. 1200–1212, Sept 2004. ISSN 1057-7149. Citations on pages 33 and 34.

DRUZHKOVA, P. N.; KUSTIKOVA, V. D. A survey of deep learning methods and software tools for image classification and object detection. **Pattern Recognition and Image Analysis**, v. 26, n. 1, p. 9–15, Jan 2016. ISSN 1555-6212. Available: <https://doi.org/10.1134/S1054661816010065>. Citation on page 27.

FERNÁNDEZ, C.; LLORCA, D. F.; STILLER, C.; SOTELO, M. A. Curvature-based curb detection method in urban environments using stereo and laser. In: **2015 IEEE Intelligent Vehicles Symposium (IV)**. [S.l.: s.n.], 2015. p. 579–584. ISSN 1931-0587. Citation on page 23.

FUKUSHIMA, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. **Biological Cybernetics**, v. 36, n. 4, p. 193–202, Apr 1980. Available: <https://doi.org/10.1007/BF00344251>. Citation on page 22.

GEIGER, A.; LENZ, P.; STILLER, C.; URTASUN, R. Vision meets robotics: The kitti dataset. **International Journal of Robotics Research (IJRR)**, 2013. Citations on pages 51 and 52.

GEIGER, A.; LENZ, P.; URTASUN, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In: **Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2012. Citation on page 51.

GEIGER, A.; ROSER, M.; URTASUN, R. Efficient large-scale stereo matching. In: KIMMEL, R.; KLETTE, R.; SUGIMOTO, A. (Ed.). **Computer Vision – ACCV 2010**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 25–38. ISBN 978-3-642-19315-6. Citation on page 51.

GOODFELLOW, I.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. Generative adversarial nets. In: GHAHRAMANI, Z.; WELLING, M.; CORTES, C.; LAWRENCE, N. D.; WEINBERGER, K. Q. (Ed.). **Advances in Neural Information Processing Systems 27**. Curran Associates, Inc., 2014. p. 2672–2680. Available: <http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>. Citation on page 27.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A. Improved Training of Wasserstein GANs. **arXiv e-prints**, p. arXiv:1704.00028, Mar. 2017. Citations on pages 30 and 31.

HARMS, H.; BECK, J.; ZIEGLER, J.; STILLER, C. Accuracy analysis of surface normal reconstruction in stereo vision. In: IEEE. **Intelligent Vehicles Symposium Proceedings, 2014 IEEE**. [S.l.], 2014. p. 730–736. Citation on page 42.

HERVIEU, A.; PAPADAKIS, N.; BUGEAU, A.; GARGALLO, P.; CASELLES, V. Stereoscopic image inpainting: Distinct depth maps and images inpainting. In: **2010 20th International Conference on Pattern Recognition**. [S.l.: s.n.], 2010. p. 4101–4104. ISSN 1051-4651. Citation on page 33.

IIZUKA, S.; SIMO-SERRA, E.; ISHIKAWA, H. Globally and Locally Consistent Image Completion. **ACM Transactions on Graphics (Proc. of SIGGRAPH)**, v. 36, n. 4, p. 107:1–107:14, 2017. Citations on pages 34, 39, 40, and 53.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: PEREIRA, F.; BURGESS, C. J. C.; BOTTOU, L.; WEINBERGER, K. Q. (Ed.). **Advances in Neural Information Processing Systems 25**. Curran Associates, Inc., 2012. p. 1097–1105. Available: <<http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>>. Citation on page 22.

Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, Nov 1998. ISSN 1558-2256. Citations on pages 22 and 27.

LIPTON, Z. C.; BERKOWITZ, J.; ELKAN, C. **A Critical Review of Recurrent Neural Networks for Sequence Learning**. 2015. Citation on page 27.

Matias, L. P. N.; Sons, M.; Souza, J. R.; Wolf, D. F.; Stiller, C. Veigan: Vectorial inpainting generative adversarial network for depth maps object removal. In: **2019 IEEE Intelligent Vehicles Symposium (IV)**. [S.l.: s.n.], 2019. p. 310–316. Citation on page 62.

MATIAS, L. P. N.; SOUZA, J. R.; WOLF, D. F. **Environment reconstruction on depth images using Generative Adversarial Networks**. 2019. Citation on page 62.

NAKAGAWA, Y.; UCHIYAMA, H.; NAGAHARA, H.; TANIGUCHI, R. Estimating surface normals with depth image gradients for fast and accurate registration. In: **2015 International Conference on 3D Vision**. [S.l.: s.n.], 2015. p. 640–647. Citations on pages 42, 43, and 44.

OLKIN, I.; PUKELSHEIM, F. The distance between two random vectors with given dispersion matrices. **Linear Algebra and its Applications**, v. 48, p. 257 – 263, 1982. ISSN 0024-3795. Available: <<http://www.sciencedirect.com/science/article/pii/0024379582901124>>. Citation on page 29.

Pan, Z.; Yu, W.; Yi, X.; Khan, A.; Yuan, F.; Zheng, Y. Recent progress on generative adversarial networks (gans): A survey. **IEEE Access**, v. 7, p. 36322–36333, 2019. ISSN 2169-3536. Citation on page 27.

PETRESCU, L.; MORAR, A.; MOLDOVEANU, F.; MOLDOVEANU, A. Kinect depth inpainting in real time. In: **2016 39th International Conference on Telecommunications and Signal Processing (TSP)**. [S.l.: s.n.], 2016. p. 697–700. Citation on page 36.

RAWAT, W.; WANG, Z. Deep convolutional neural networks for image classification: A comprehensive review. **Neural Computation**, v. 29, n. 9, p. 2352–2449, 2017. PMID: 28599112. Available: <https://doi.org/10.1162/neco_a_00990>. Citation on page 27.

Ridel, D.; Rehder, E.; Lauer, M.; Stiller, C.; Wolf, D. A literature review on the prediction of pedestrian behavior in urban scenarios. In: **2018 21st International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2018. p. 3105–3112. Citation on page [22](#).

SHEN, J.; CHEUNG, S.-C. S. Layer depth denoising and completion for structured-light rgb-d cameras. **2013 IEEE Conference on Computer Vision and Pattern Recognition**, p. 1187–1194, 2013. Citation on page [35](#).

SWAIN, M. J.; BALLARD, D. H. Indexing via color histograms. In: SOOD, A. K.; WECHSLER, H. (Ed.). **Active Perception and Robot Vision**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992. p. 261–273. ISBN 978-3-642-77225-2. Citation on page [56](#).

TAN, C.; SUN, F.; KONG, T.; ZHANG, W.; YANG, C.; LIU, C. A survey on deep transfer learning. In: KÚRKOVÁ, V.; MANOLOPOULOS, Y.; HAMMER, B.; ILIADIS, L.; MAGLOGIANNIS, I. (Ed.). **Artificial Neural Networks and Machine Learning – ICANN 2018**. Cham: Springer International Publishing, 2018. p. 270–279. Citation on page [27](#).

UHRIG, J.; SCHNEIDER, N.; SCHNEIDER, L.; FRANKE, U.; BROX, T.; GEIGER, A. Sparsity invariant cnns. **2017 International Conference on 3D Vision (3DV)**, p. 11–20, 2017. Citations on pages [13](#), [36](#), and [37](#).

WEERASEKERA, C. S.; DHARMASIRI, T.; GARG, R.; DRUMMOND, T.; REID, I. Just-in-time reconstruction: Inpainting sparse maps using single view depth predictors as priors. In: **2018 IEEE International Conference on Robotics and Automation (ICRA)**. [S.l.: s.n.], 2018. p. 1–9. ISSN 2577-087X. Citation on page [35](#).

XUE, H.; ZHANG, S.; CAI, D. Depth image inpainting: Improving low rank matrix completion with low gradient regularization. **IEEE Transactions on Image Processing**, v. 26, n. 9, p. 4311–4320, Sept 2017. ISSN 1057-7149. Citation on page [37](#).

YANG, S.; HUANG, Y.; SCHERER, S. Semantic 3d occupancy mapping through efficient high order crfs. **2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**, p. 590–597, 2017. Citation on page [23](#).

YU, J.; LIN, Z.; YANG, J.; SHEN, X.; LU, X.; HUANG, T. S. Generative image inpainting with contextual attention. **arXiv preprint arXiv:1801.07892**, 2018. Citations on pages [14](#), [15](#), [34](#), [40](#), [41](#), [46](#), [52](#), [53](#), [55](#), [57](#), [58](#), and [59](#).

ZHANG, L.; SHEN, P.; ZHANG, S.; SONG, J.; ZHU, G. Depth enhancement with improved exemplar-based inpainting and joint trilateral guided filtering. In: **2016 IEEE International Conference on Image Processing (ICIP)**. [S.l.: s.n.], 2016. p. 4102–4106. ISSN 2381-8549. Citation on page [36](#).

ZHANG, Y.; FUNKHOUSER, T. Deep depth completion of a single rgb-d image. **The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, 2018. Citation on page [37](#).