Fully residual Unet-based semantic segmentation of automotive fisheye images: a comparison of rectangular and deformable convolutions

Rosana El Jurdi, Ahmed Rida Sekkat, Yohan Dupuis, Pascal Vasseur, Paul Honeine

Fully Residual Unet-based Semantic Segmentation of Automotive Fisheye Images : a Comparison of Rectangular and Deformable Convolutions

Rosana EL Jurdi[a,d], Ahmed Rida Sekkat[a,e], Yohan Dupuis[b], Pascal Vasseur[c], Paul Honeine[a]

[a] *LITIS, University of Rouen Normandy, Rouen, France*
[b] *CESI LINEACT, CESI, France*
[c] *MIS, University of Picardie Jules Verne, Amiens, France*
[d] *Institut du Cerveau - Paris Brain Institute, Sorbonne University, Paris, France*
[e] *IAV GmbH, Berlin, Germany*

Abstract

Semantic image segmentation is an essential task for autonomous vehicles and self-driving cars where a complete and real-time perception of the surroundings is mandatory. Convolutional Neural Network approaches for semantic segmentation standout over other state-of-the-art solutions due to their powerful generalization ability over unknown data and end-to-end training. Fisheye images are important due to their large field of view and ability to reveal information from broader surroundings. Nevertheless, they pose unique challenges for CNNs, due to object distortion resulting from the Fisheye lens and object position. In addition, large annotated Fisheye datasets required for CNN training is rather limited. In this paper, we investigate the use of Deformable convolutions in accommodating distortions within Fisheye image segmentation for fully residual U-net by learning unknown geometric transformations via variable shaped and sized filters. The proposed models and integration strategies are exploited within two main paradigms: single(front)-view and multi-view Fisheye images

*Corresponding authors:
**This work was done while Ahmed Rida Sekkat was with LITIS, University of Rouen Normandy.
Email addresses: rosana.eljurdi@icm-institute.org (Rosana EL Jurdi), ahmed.rida.sekkat@iav.de (Ahmed Rida Sekkat), ydupuis@cesi.fr (Yohan Dupuis), pascal.vasseur@u-picardie.fr (Pascal Vasseur), paul.honeine@univ-rouen.fr (Paul Honeine)

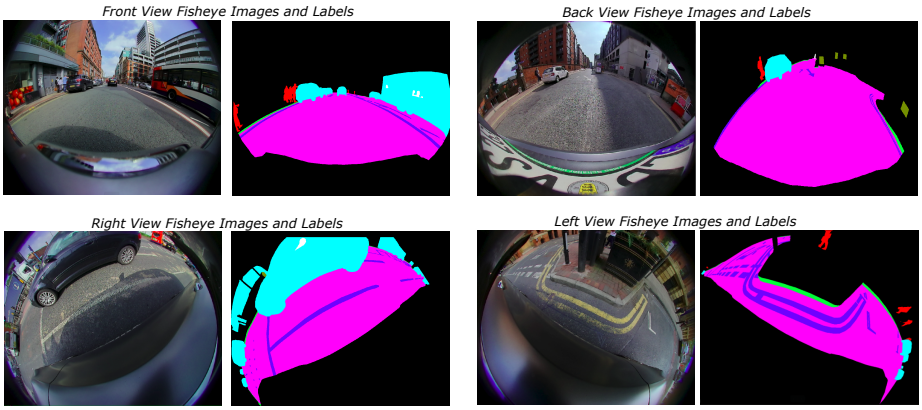


Figure 1: WoodScape dataset image examples and their labels from four views: TOP LEFT: Front-view, TOP RIGHT: Back-view, BOTTOM LEFT: Right-view, BOTTOM RIGHT: Left-view

segmentation. The validation of the proposed methods is conducted on synthetic and real Fisheye images from the WoodScape and the SynWoodScape datasets. The results validate the significance of the Deformable fully residual U-Net structure in learning unknown geometric distortions in both paradigms, demonstrate the possibility in learning view-agnostic distortion properties when trained on the multi-view data and shed light on the role of surround-view images in increasing segmentation performance relative to the single view. Finally, our experiements suggests that Deformable convolutions are a powerful tool that can increase the efficiency of fully residual U-Nets for semantic segmentation of automotive fisheye images.

Keywords: Fisheye Image Segmentation, Multi-view Data Augmentation, Deformable Convolutions, Deep Convolutional Neural Networks

---

1. Introduction

Semantic segmentation is defined as the process of pixel-wise labeling of images in order to extract important objects, such as pedestrians, road lanes,

buildings, traffic signals etc, while incurring detection tasks at the same time. In the essence of autonomous driving, images acquired via Fisheye cameras are useful as they capture large areas of surrounding scenes thanks to their broad field of view. Fisheye data are of particular importance, as they pave the way for safer automotive low speed manoeuvring such as parking, collision avoidance, and right turn resistance where an accurate full coverage is required. As a result, they provide valuable information given many applications and allow for a proficient autonomous scene understanding.

Semantic segmentation solutions via Convolutional Neural Networks (CNNs) standout over other state-of-the-art solutions, due to the ability of CNNs to be trained end-to-end, as well as their powerful generalization ability over new data. CNNs are significant as they allow the modeling of prior knowledge regarding geometric transformations thanks to their model capacity, and translational invariance modules (e.g. max pooling layers) [8]. Due to the current availability of high performance Graphics Processing Units (GPUs) and excellent open source deep learning frameworks, CNN-based solutions for semantic segmentation have registered a breakthrough in different applications with the field of autonomous driving being no exception to the rule [14, 35, 27].

A pioneering approach for image segmentation is the U-Net model [25], that is a symmetric encoder/decoder structure with skip connections. The encoder part is a contracting path composed of stacked convolutional and max pooling layers, whereas the decoder part is an expanding path composed of deconvolutional or bilinear upsampling layers. Layers within the encoder are dedicated to capturing contextual information in order to detect objects/classes present in an image. On the other hand, the decoder layers help precise localization of patterns, thus indicating where in the image an object is located. As an image moves further into the contracting layers, it decreases in size but increases in depth of its learnt contextual features. In contrast, the decoder layers increase size but decrease its depth, thus retaining the model's localization ability. Skip connections combine symmetrical contextual and positional features from opposing convolutions in the two corresponding paths. In addi-

tion to concatenating features from corresponding encoder/decoder layers [25], skip connections could also be used to combine features from consecutive layers within the same encoder/decoder parts. A very well-known structure that makes use of inter-layer nested skip connections is the Residual-Unet proposed in [23]. The nested connections combine features from different layers of the encoder parts (or decoder parts), thus, evading the deterioration of information throughout the internal layers of the networks.

1.1. Related works on semantic segmentation of large field of view automotive images

Compared to perspective images, segmentation of Fisheye images via traditional CNNs encounter several challenges. A major limitation of CNNs is that they are highly dependent on the existence of large-scale annotated training Fisheye datasets to gain their generalization ability. Till now, there is a scarcity in the public fully-annotated datasets of Fisheye images dedicated to road scene understanding[1]. Moreover, acquiring and annotating such a dataset is rather expensive and laborious. Up until our knowledge, there have been only three public datasets for Fisheye images with semantic segmentation ground truths. The OmniScape dataset [31], the WoodScape dataset, and the SynWoodScape dataset [30].

Another limitation for CNNs is their ability to model new or unknown geometric transformations. CNNs can learn some transformations known to the user such as an object position or orientation. However, these approaches are limited in their ability to model new tasks with variable or unknown geometric properties [8]. In Fisheye images, these drawbacks are particularly persistent due to the distorted nature of objects in the image depending on the view angle of the object acquired relative to the Fisheye camera. Thus, there are countable limitations to traditional CNNs generalization ability because of large non-linear distortion [36].

[1] https://sites.google.com/view/omnicv2022/useful-datasets

Some studies have addressed these limitations by reconstructing the Fisheye lens using data augmentation techniques on perspective image datasets, or by extracting omnidirectional images from simulators in order to allow the learning of the underlying geometric representation in the images. For instance, several works used a tangent transformation on perspective datasets to simulate the Fisheye lens [27, 9, 26]. Other studies used the resulting transformed images with different architectures based on planar conventional convolutions [10, 26, 24]. Despite their significance, one could point out that the generated images are not as rich in information and do not hold the same field of view as the real Fisheye data. This sheds light on the importance of proposing alternative methods to data augmentation, where the augmented images share similar acquisition and surrounding properties as the target dataset. On the other hand, other researchers proposed methods based on the spherical representation of omnidirectional images, such as equirectangular, panoramic or spherical representations based on the icosahedron subdivision to model a sphere [29]. These methods can be adapted to Fisheye images since an equirectangular image can be considered as an intermediate representation of a Fisheye image.

Contrastingly, Sekkat et al. proposed in [31] a framework that simulates real omnidirectional images using their calibration parameters. The authors generated Fisheye images from a virtual hyper-realistic open-world game (GTA-V) simulating a real city. This framework was extended in [31] by the same authors with OmniScape, a synthetic dataset using both GTA-V and CARLA simulators, the latter being an open-source simulator for autonomous driving research.

The use of synthetic datasets can be limited when dealing with the segmentation of real data. This essentially depends on the realistic textures that can be generated from the simulator [29]. Moreover, synthetic Fisheye data can only be generated via known calibration parameters. In real Fisheye datasets, each camera has unique calibration parameters. As a result, the deformation of objects can be different from one camera to another. Moreover, the degree of distortion of an object depends on the Fisheye acquisition camera position,

orientation, object position and the field of view. As a result, segmenting objects within Fisheye data is rather subjected to multiple challenges caused by the variability in acquisition parameters and as a result the degree of object distortion. In this paper, we are interested in proposing a geometry-agnostic method that is able to learn the distortions produced by an omnidirectional camera directly from the resulting real images.

One approach to learning unknown geometric transformations is via Deformable convolutions as proposed by [8]. Deformable convolutions allow learning geometric deformations customized to each dataset while training. Thus, instead of fixed kernel sizes over all the network layers, the method proposes learnable size and shape kernels. The variable kernel size effect is generated by shifting the regular sampling locations by a 2D offset thanks to an additional convolutional layer learned end-to-end with the main convolutions in the network. Deformable convolutions have shown promising potential for object detection and segmentation tasks given perspective images [8]. Due to their powerful ability in modeling geometric transformations, persistent in Fisheye data, these components have rightfully raised interest regarding their ability to accommodate Fisheye geometric characteristics as explored in [9, 21].

Ahmed and Lecue [1] demonstrated that learning the shape of convolution kernels in non-Euclidean hyperbolic spaces is better than deformable kernel methods, but the proposed method was not tested in real Fisheye images. Playout et al. [22] proposed an adaptation protocol to adapt models trained on perspective images to Fisheye images using deformable convolutions. Hu et al. [13] proposed a semantic segmentation network dedicated for panoramic images of outdoor scenes based on a distortion convolutional module that aims to correct the image deformation. Nevertheless, the real added value of the Deformed convolutional has not yet been assessed within U-Net like structures for Fisheye image segmentation.

The main contributions of the paper lie within the scope of investigating Deformable convolutions as a proficient substitute to convolutional layers for Fisheye image segmentation for fully residual U-Net, in both front-view and

multi-view scene processing. In addition, we also explore the tendency of images from different views to ameliorate segmentation performance and the efficiency of the U-net variants. Finally, we highlight the role of Deformable convolutions in aiding view-agnostic learning. We note that the objective is not dedicated to achieving better than state-of-the-art results, but to shed light on the true added value of Deformable convolutions in U-Net like models for Fisheye image segmentation. This paper also provide baseline results on the newly released Synwoodscape dataset.

The rest of the paper is organized as follows. Section 2 presents the concept of Deformable convolutions and provides brief overviews of two important public Fisheye datasets, the WoodScape and the SynWoodScape datasets. Section 3 elaborates on the proposed Deformable Residual-Unet model, as well as the explored multiple frameworks and integration strategies. Section 4 evaluates the relevance of the proposed Deformed Residual-Unet on several datasets. Finally, Section 5 concludes this paper with future works.

2. Preliminaries

2.1. Deformable Convolutions Concept

Deformable convolutions are convolutional layer variants that allow learning of unknown geometric transformations via variable shaped and sized kernels/-filters, rather than fixed sized convolutions over the entire network structure [8]. Instead of customizing the kernels as adopted by many state-of-the-art approaches [15, 6, 16], the variable shaped filter effect is simulated via adding 2D offsets to the regular sampling locations. The novel locations are obtained in correspondence with the geometric properties of the input via the addition of a fractional offset to the original input followed by a bilinear interpolation. To generate the fractional offsets, an additional convolutional layer is added to the network and trained simultaneously with the traditional convolutional layer of the model. In such a way, the kernel shapes and sizes are learned to accommodate the unknown deformations particular to each dataset. The offset

convolution filter has the same filter size as that of the regular convolutions and the same stride. The offset convolutional filter takes the original input sample and produces an output of the same spatial resolution. The output indicates the desired fractional offsets that are to be applied to each pixel in the input conforming with the geometrical distortion. The integer positions of the new sampled inputs are obtained via bilinear interpolation. The final input to the original network convolutions is the new sampled input obtained by the addition of the integer offset/relative positions to the original input pixels. The process is demonstrated in Figure 4.

2.2. Benchmark datasets

In this paper, we used two well-known datasets, WoodScape [36] and its synthetic version the SynWoodScape [30]. From the WoodScape dataset, 8234 annotated images were collected asynchronously from the four different view angles of a vehicle. The semantic annotations are provided relative to 10 classes including road, lanemarks, curb, person, rider, vehicle, bicycle, motorcycle, traffic sign, in addition to the void class. Samples of the dataset are shown in Figure 1. From a closer look at the dataset characteristics as shown in Figure 2, it is realized that there is a high class size imbalance relative to the occupancy of particular classes in the entire images. Moreover, these classes are often located at the periphery of the Fisheye images, which means they are highly distorted. 8000 annotated images from the SynWoodScape dataset were used. The semantic segmentation annotation is provided for 25 classes. We choose to use 20 classes by aggregating classes, such as ego-vehicle that was included to four-wheeler vehicles. By taking a closer look at the class size distribution given in Figure 3, we realize that the dataset is characterized with a high class imbalance relative to its percentage occupancy in the images. For example, the four-wheeler vehicle class may occupy about 90% of the image area in some sample images, the water class can at most occupy less than 1% of the image.

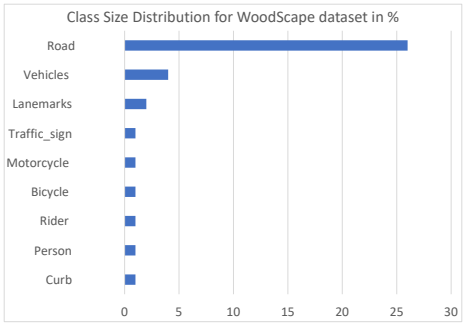


Figure 2: The WoodScape dataset class size distribution indicating the average possible size (pixel occupancy) of each class in the dataset. A high size imbalance between classes in the dataset is mainly due to the road class, which is the most prominent class in size.

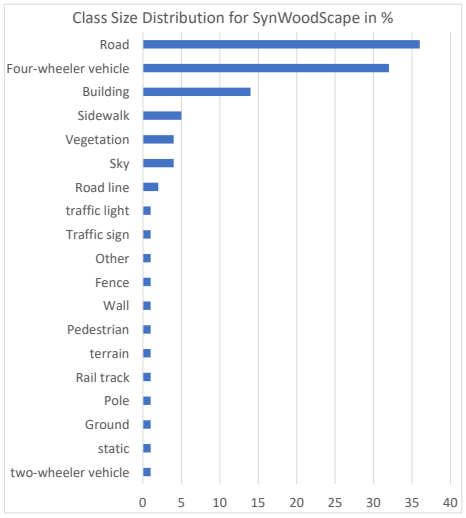


Figure 3: The SynWoodScape dataset class size distribution indicating the average size of each class in the dataset. The road, four-wheeler vehicle and building classes are the most prominent in size, indicating a high size imbalance between them and the other classes in the dataset.

3. Proposed Method

In this section, we present the concept of Deformable convolutional layers and its implementation within the segmentation framework for Fisheye datasets. We further elaborate on the proposed Deformed Residual-Unet model, the un-

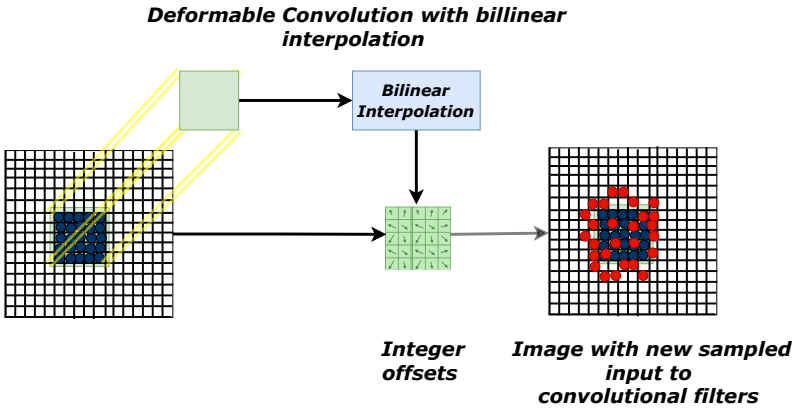


Figure 4: Deformable convolution concept. The Deformable convolution effect is generated via an additional convolutional layer (green) followed by bilinear interpolation. The convolutional filter generates fraction offsets whereas the bi-linear interpolation transforms the fractional offset to integral positions. The new sampled input is the addition of the original input (blue dots) to the offset integral positions.

derlying building blocks, and the different integration strategies and investigated paradigms.

3.1. Proposed architecture

The adopted baseline U-Net like architecture is a fully Residual-Unet as proposed by [17] based on [23]. Compared to the original implementation of the U-Net architecture, the adopted fully Residual U-Net has two main upgrades. First, instead of concatenation between encoder and decoder layers, the concatenation is replaced with addition, thus allowing the network to evade vanishing gradient problems. In addition to the long skip connections, the network also has internal nested connections between the different convolutional blocks composing the encoder and decoder layers. In this way, the network improves the flow of information and avoids deterioration of information through the internal layers of the network.

The network is a 4-stage encoder/decoder architecture with long skip connection between per-stage encoder and decoder blocks. Long skip connections combine features from each convolutional block in the encoder part with its

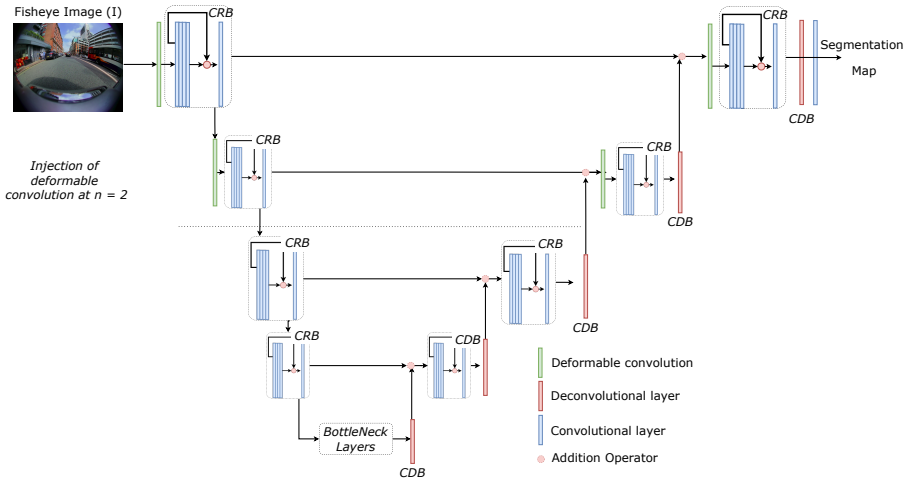


Figure 5: DRU-EnD(+2) model where the Deformable convolution is injected up to the n^{th} level of the Residual-Unet before the first layer of the Residual Convolutional block (CRB) in both the encoder and decoder parts.

corresponding equivalent in the decoder part. The Residual-Unet is constituted of two main building blocks: the convolutional residual blocks (CRB) and the convolutional decoder blocks (CDB). The Convolutional Residual blocks (CRB) is composed of 5 convolutional layers each followed by batch normalization with residual connections existing between the first convolutional layer and the 4th convolutional layer. The output from the first and 4th convolutional blocks are added and fed to the last convolutional layer. The combination of the different outputs from different convolutions per layer allows a more fine-grained feature extraction. The CDB is composed of transposed convolutions followed by batch normalization.

The encoder is composed of 4 ensembles of CRB followed each by a max pooling layer. On the other hand, the decoder is composed of CDB and CRB. The input to each CRB in the encoder is the output of the block that precedes, whereas the input to each CRB in the decoder is the addition of the corresponding CRB in the same encoder stage with that of the output of the convolutional decoder blocks in the stage preceding it.

The proposed model builds upon the architecture proposed by [23] and the Deformable convolutions in [8]. Thus, we extend upon the fully Residual U-Net by replacing convolutional blocks with the Deformable convolution according to several integration strategies. Our contribution is at the level of the first convolutional layer of the CRB. Thus, we replace the regular 2D convolution adopted by the deformed convolutional block demonstrated previously. In such a way, we allow the network to take into consideration spatial and geometric aspects while training. In the following, we will elaborate on the proposed integration strategies.

3.2. Integration Strategies

We investigate different integration strategies of the Deformable convolutional block onto the Residual-Unet Baseline. The proposed model is denoted as Deformable Residual-Unet **(DRU)**. The most simplest integration is at the input level of the Residual-Unet where we replace the first convolutional layer with that of the Deformable convolutional block. We denote this model by **DRU-L(+1)**. Alternatively, we also explore the possibility of integrating the Deformable convolutional block at the last convolutional block with kernel size greater than one in the decoder layer. We denote this model by **DRU-Dec(+1)**. Finally, **DRU-EnD(+n)** demonstrates the integration of Deformable convolutions up to the n^{th} stage of the Residual-Unet, i.e., in the n^{th} convolutional block of the encoder and its corresponding convolutional block at the decoder. Experiments were carried out with $n = 1$, 2 and 3. The corresponding models are denoted as **DRU-EnD(+1)**, **DRU-EnD(+2)**, and **DRU-EnD(+3)**.

3.3. Explored Paradigms

We investigate the proficiency of Deformable-Unet given three main paradigms: **Front-view train front-view test**, **multi-view train multi-view test** and **multi-view train front-view test**. In the state of the art, datasets dedicated for autonomous driving like CityScapes [7] and CamVid [3]

contain images from just the front-view front cameras. In this essence, we focus on the first paradigm, i.e., **front-view train front-view test**, on conducting training via just the front-view image for WoodScape and SynWoodScape similar to the works of [4, 7].

In addition to front-view, we also explore the adaptability of the proposed model given multi-view Fisheye image segmentation. The main intuition is to explore the possibility of a view-agnostic model that can generalize well by considering the different information and learnt features from multiple views. We note that, in this paradigm, left and right Cameras in the WoodScape and SynWoodScape datasets are facing down resulting in very specific images where a large amount of pixels represent the road and the ego-vehicle. This paradigm, denoted **multi-view train multi-view test**, is similar to the most recent state-of-the-art works that aim to learn on acquired Fisheye images. This is demonstrated by the different models and experiments proposed in [36].

In the last paradigm, denoted **multi-view train front-view test**, we address the possibility of exploiting images from the different surround views in order to increase the segmentation performance on single front-view data. Up to our knowledge, this idea is novel to the state of the art as it endorses data augmentation via the different image views from the four Fisheye cameras. Our code and models can be found in this github repository [2].

3.4. Computation Complexity and Model Parameters

In order to evaluate the computational complexity of the proposed models, we compute the amount of multiply–accumulate operations (MAC) following the flops counter described in the GitHub repository[3]. The computational complexity is measured in Giga MACs and the number of parameters in Millions. Results are Benchmarked in Table 1 given an input of dimensions 512×512. The Baseline model corresponds to the fully residual U-Net with rectangular

[2] https://github.com/rosanajurdi/WoodScape-Segmentation-Project
[3] https://github.com/sovrasov/flops-counter.pytorch

Models	Computational Complexity (Giga MACs)	Model Parameters (Millions)
Baseline	125.20	19.65
DRU-L(+1)	125.39	19.65
DRU-Dec(+1)	127.23	19.66
DRU-EnD(+1)	127.42	19.65
DRU-EnD(+2)	128.95	19.68
DRU-EnD(+3)	129.71	19.73

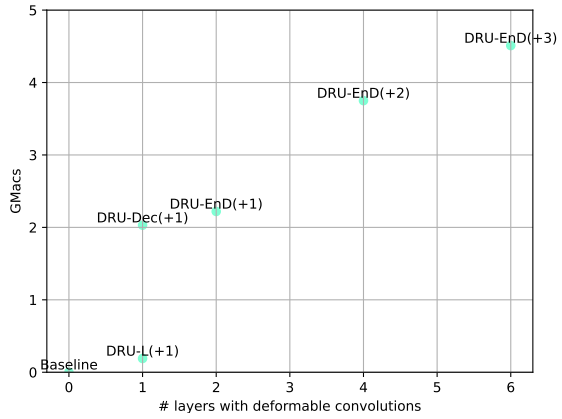
Table 1: Computational Complexity measured in multiply–accumulate operations (Giga MACs) and Model Parameters in Millions, for the integration strategies of the Deformable Convolutional layer.

convolutions. These results show that the integration of Deformable convolutions into the network adds only a small overhead on model parameters and computation. Figure 6 presents the overhead, relative to the baseline model, for different values of the number of layers where Deformable convolutions are applied. The Deformable convolution seems to have a slightly bigger overhead when applied on the deconvolution layer. The size increase in the deconvolution layer explains this result. It can also be noticed that the increase in computational complexity seems linear while the number of parameters increases in a less linear fashion.

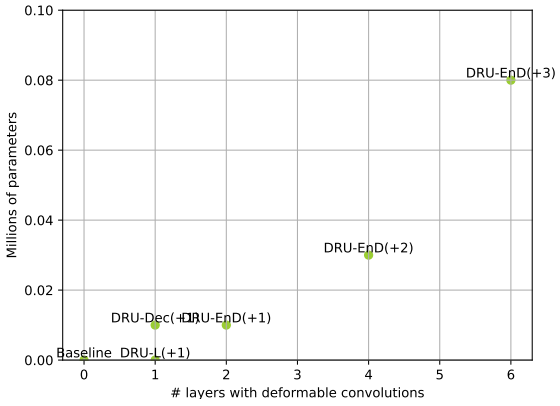
3.5. Connections to related work

Deformable Convolutions

Due to their powerful ability in modeling geometric transformations, several works have extended upon the concept of Deformable convolutions in order to adapt CNN methods to Fisheye data [9, 21]. For instance, the authors of [9] proposed the concept of Restricted Deformable Convolution where the central location of the filters are rather fixed while the other locations are learned via the convolutional mapping layer. Alternatively, the authors of [21] proposed to deploy the Deformable convolutions on top of CNNs pre-trained on perspective



(a) Compute complexity



(b) Parameter set size

Figure 6: Deformable convolution overheads, relative to the Baseline set at $(0, 0)$.

images and to finetune the structure via a tiny sample of annotated Fisheye images. Moreover, they investigated the minimal number of samples needed to adapt traditional CNNs on Fisheye images via the concept of Deformable convolutions. Despite their significance, however, up to our knowledge, none of these works adopted or investigated the validity of exploiting the Deformable convolutional component within U-Net like structures. Evidently, the U-Net is a very well-known and adopted segmentation model. The contribution of this work is driven by the importance of having deployment mechanisms able to accommodate objects with different scales and deformations at the higher level

layers of the CNNs. These layers are mainly dedicated to encoding semantic features over spatial locations as noted in [8]. In a U-Net or a U-Net like structure, this relationship is imposed by the skip connections existing between the Encoder and Decoder layers. The main intuition behind the work is that the addition of the Deformable component at corresponding convolutions per-stage in a U-Net can sufficiently increase the networks ability to learn geometric features specific to the Fisheye images and their characteristics, therefore enhancing the segmentation performance.

Data augmentation

In comparison to the state of the art, works of [2] and [11] exploit data augmentation methods that can be considered as random rotated cropping with the constraint of using the omnidirectional representation, either by using a Fisheye calibration model, or a 3D representation of a sphere. For instance, the tangent images proposed in [11] could be thought of as a data augmentation method based on dedicated cropping relative to a plane tangent to the icosahedron representation of a sphere. Despite their significant role in transforming omnidirectional data to perspective data with low distortions, this method is more useful for high-resolution equirectangular images. Moreover, it results in multiple crops with redundancy, since the same object in the scene can be included in multiple crops, leading to more computational time to segment the same objects in the scene. Moreover, sub-sectioning spherical data into perspective ones with low distortion, may result in information loss particularly relative to the global position of the objects of interest. We argue that our proposed augmentation method preserves the holistic spatial understanding of the Fisheye images lost via random cropping. This will in turn positively impact the segmentation performance as the Fisheye image properties are preserved and learnt through training. In this paper, we propose to exploit surround view data for augmentation in order to preserve the holistic spatial understanding and properties of a Fisheye image. We argue that the exploitation of multi-view data augmentation allows better feature representation particular to the Fisheye

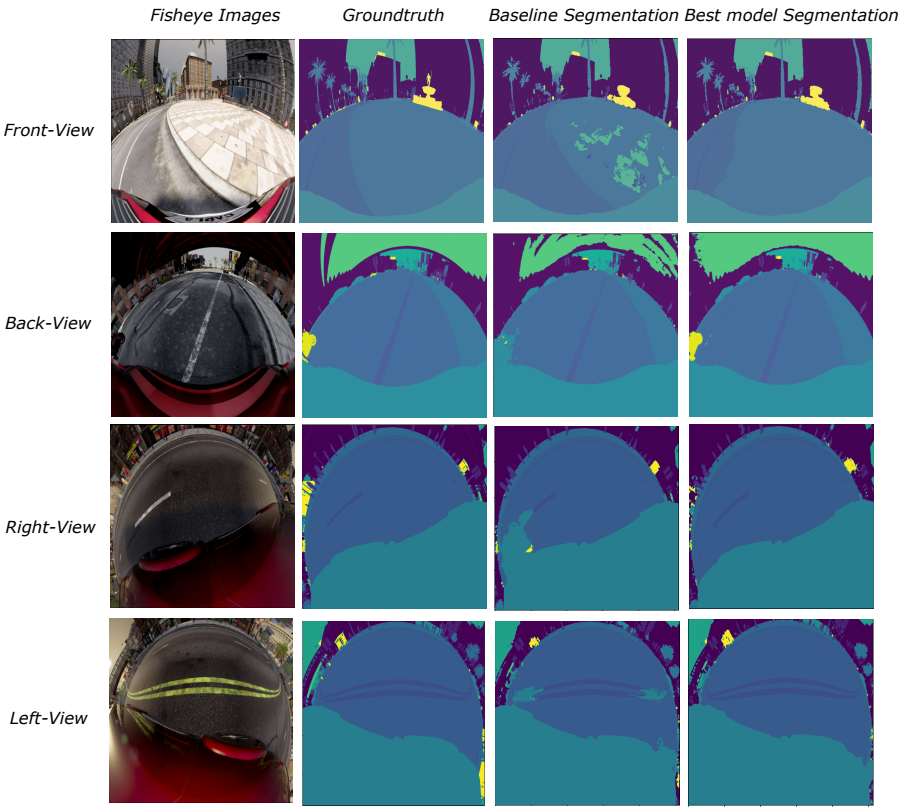


Figure 7: Qualitative Results on the SynWoodScape dataset: Fisheye Image, Ground-truth, Baseline (Residual Unet segmentation) and Best model performance (DRU-EnD(+3))

data at hand.

4. Experimental results

Experiments are conducted on the WoodScape dataset of real images[4] and the SynWoodScape dataset [30] of synthetic images generated using the same calibration parameters as the real ones. In the following, we first describe the experimental setup and analyze the results on the multiple paradigms.

[4] https://woodscape.valeo.com/home

Table 2: Mean Intersection over Union (mIoU) results on the SynWoodScape dataset (Synthetic images) when trained and tested on single-view images, validated over 3 Monte Carlo Simulations

	Residual-Unet	DRU-L(+1)	DRU-Dec(+1)	DRU-EnD(+1)	DRU-EnD(+2)	DRU-EnD(+3)
Building	88.06± 3.08	87.81± 1.96	88.36± 1.86	89.92± 0.31	90.23± 1.09	90.51± 0.28
Fence	2.38± 3.33	2.11± 2.90	0.32± 0.33	9.90± 8.10	8.05± 5.19	14.80± 6.76
Four-wheeler vehicle	83.38±18.84	93.51± 4.19	92.66± 1.82	93.38± 5.49	95.72± 3.26	97.45± 0.06
Ground	58.40±23.39	54.42± 7.53	64.43± 7.15	71.50± 3.84	58.04±18.21	79.39± 2.90
Other	56.30±16.25	60.36±11.35	61.11±12.29	68.03± 1.39	66.57± 1.74	66.00± 1.52
Pedestrian	44.06±20.59	47.34±13.10	46.64±10.40	55.11± 2.71	58.71± 1.56	58.52± 3.93
Pole	31.85±10.62	33.33± 9.25	30.98± 2.00	39.98± 2.56	42.32± 5.44	44.89± 5.14
Rail track	46.42±27.43	11.12±16.97	41.03±26.84	58.26±28.01	73.37±17.52	80.47±12.02
Road	91.68± 9.06	95.11± 2.58	96.09± 0.40	96.55± 2.54	97.51± 1.46	98.31± 0.11
Road line	74.04±16.69	78.69±10.87	77.80± 4.27	84.54± 1.62	86.37± 2.32	87.00± 2.30
Sidewalk	76.01±17.29	74.42± 4.66	78.16± 0.93	83.93± 6.45	85.62± 7.75	89.78± 1.18
Sky	93.73± 1.92	94.34± 1.57	94.12± 0.95	95.16± 0.53	95.16± 0.20	95.14± 0.16
Static	38.95±10.53	34.70±14.62	38.18± 3.27	42.79± 1.68	49.14± 8.00	54.87± 1.49
Terrain	2.23± 1.88	3.37± 3.68	3.66± 4.95	2.02± 1.59	8.18± 3.00	3.60± 2.57
Traffic light	46.03±25.09	52.66±14.85	51.59±12.27	62.86± 2.96	56.48± 3.32	61.40± 0.21
Traffic sign	21.70±17.07	28.26±15.61	26.41±15.77	39.38± 4.46	37.76± 2.62	41.15± 3.06
Two-wheeler vehicle	14.97± 1.72	19.34± 7.00	18.34± 1.67	22.43± 7.14	21.45±10.24	19.88± 3.00
Vegetation	65.30±18.86	71.16±11.91	69.90±11.88	77.49± 0.72	78.20± 2.28	78.97± 0.91
Wall	27.98±31.23	36.70±23.70	33.45±19.18	52.96± 8.02	58.09± 9.11	59.92± 3.70
Water	17.22±24.09	16.28±14.05	17.81±16.03	29.40± 5.40	24.91± 9.34	33.72± 2.67
Average	44.41±17.47	45.76±10.05	47.81± 6.31	55.98± 1.44	56.76± 4.97	**59.80± 0.39**

4.1. Settings

To insure reproducibility, we deployed the experimental framework and fully Residual-Unet model presented in [17]. The Deformable convolutions can be found in this github repository[5]. For the loss, we use the cross entropy loss. Models were initialized randomly and trained from scratch. Thus, we do not use pre-trained layers. The methods were evaluated using the mean Intersection over Union (mIoU). Training was conducted via the Adam optimizer with a batch size of 2 over 45 epochs. The learning rate was set to 5×10^{-4} and halved each 20 epochs if the validation performance did not improve.

For pre-processing, we have resized the RGB images to a size of 512 × 512 and normalized them to a pixel value between 0 and 1. The datasets were split into train and validation based on an (80 %, 20 %) partition respectively. Cross-validation was done on three folds of the data and results were averaged over

[5] https://github.com/4uiiurz1/pytorch-deform-conv-v2/blob/master/deform_conv_v2.py

three Monte-Carlo simulations. Our code is publically available on GitHub [6].

4.2. Front-view Fisheye Image Segmentation

In this section, we present results for the different integration strategies of the Deformable convolution on the fully Residual-Unet model when trained on single front-view images from both the synthetic and real datasets via the cross entropy loss. We note that the real dataset includes 9 classes whereas the synthetic dataset contains 20 classes. In order to establish a baseline performance, we train the traditional fully Residual-Unet in [17] via the same training strategy as the proposed models.

The results obtained on mean Intersection over Union (mIoU) on both datasets, shown in Table 2 and Table 4, indicate the significance of the proposed method for Fisheye datasets. Thus, the integration of the Deformable component at corresponding convolutional layers from the encoder and decoder paths, as the case of DRU-EnD(+1), DRU-EnD(+2), and DRU-EnD(+3), increases the average mIoU significantly relative to the Residual-Unet baseline model. In fact, it is evident from both tables that, as the number of injected Deformable components increases, the segmentation performance relative to Fisheye data increases as well. This indicates the ability of the Deformable convolutions in accommodating intrinsic Fisheye characteristics and geometric transformations specific to a particular view, here the front-view. Thus, one can say that, as the number of Deformable components increases per level ensemble encoder/decoder layers, the model is then further able to capture the intrinsic geometric distortions properties dependent or related to the camera pose and location. This paves the way to the possibility of deploying a single model for single-view segmentation given computation constraints.

4.3. Multi-view Fisheye Image Segmentation

In addition to front-view, we also present results for the different integration strategies of the Deformable convolutions when trained on multi-view Fisheye

[6] https://github.com/rosanajurdi/WoodScape-Segmentation-Project

Table 3: Mean Intersection over Union (mIoU) results on SynWoodScape dataset (Synthetic images) when trained and tested on multi-view images, validated over 3 Monte Carlo Simulations.

	Residual-Unet	DRU-L(+1)	DRU-Dec(+1)	DRU-EnD(+1)	DRU-EnD(+2)	DRU-EnD(+3)
Building	90.81± 0.98	91.76± 0.78	91.97±0.14	91.53± 0.64	91.47± 0.20	91.49± 0.20
Fence	8.88± 3.83	16.99± 1.11	18.87±0.94	17.69± 5.58	19.34± 2.67	19.49± 3.14
Four-wheeler Vehicle	97.42± 1.20	98.49± 0.87	98.99±0.08	98.34± 0.61	98.82± 0.09	98.73± 0.05
Ground	65.5± 7.54	79.67±18.76	90.21±0.22	78.99± 9.46	81.72±13.17	89.15± 1.44
Other	72.92± 2.99	75.73± 1.25	74.95±0.38	75.48± 0.51	71.67± 1.08	71.74± 0.94
Pedestrian	58.32± 3.36	62.53± 5.23	65.45±1.28	64.37± 1.67	62.38± 1.79	62.88± 0.90
Pole	36.57± 6.20	46.24± 4.10	48.59±1.46	46.81± 3.22	44.89± 4.44	45.62± 1.67
Rail track	73.10±19.85	76.51±18.01	89.06±2.34	73.25±15.19	88.80± 0.81	87.87± 0.55
Road	97.07± 0.89	98.17± 0.85	98.91±0.18	98.42± 0.52	98.64± 0.45	98.59± 0.27
Road line	84.4± 2.21	87.71± 3.84	89.31±1.64	88.11± 1.50	88.94± 0.66	89.00± 0.48
Sidewalk	78.14± 7.58	84.70± 7.90	92.10±1.02	87.56± 4.89	89.46± 4.86	89.88± 2.07
Sky	94.48± 0.25	94.72± 0.61	94.91±0.01	94.80± 0.23	94.67± 0.20	94.73± 0.16
Static	53.45± 5.75	63.46± 5.34	69.13±0.36	62.35± 6.04	65.98± 1.83	65.84± 0.68
Terrain	5.14± 4.24	13.03± 5.06	13.74±7.10	9.04± 4.97	10.07± 5.94	10.08± 4.06
Traffic light	63.53± 3.09	68.62± 1.50	68.89±0.15	69.04± 1.53	65.18± 0.60	65.24± 1.85
Traffic sign	44.04± 5.05	51.28± 1.21	50.55±1.87	50.41± 2.69	45.74± 2.10	46.55± 2.99
Two-wheeler vehicle	43.86± 1.81	51.45± 6.45	54.17±1.04	51.96± 2.16	51.24± 3.48	43.21± 3.96
Vegetation	79.75± 1.13	81.11± 0.40	81.14±0.31	80.82± 0.54	80.61± 0.31	80.68± 0.22
Wall	56.68± 6.09	66.99 ± 4.62	66.11±0.82	63.54± 4.56	65.19± 2.62	64.49± 2.93
Water	35.55± 5.77	36.91 ± 3.92	32.07±0.18	33.82± 7.74	38.07± 2.76	37.82± 2.37
Average (Multi-View)	61.98± 3.77	67.30± 4.46	69.46±0.02	66.82± 3.38	67.64± 2.28	**67.65±0.63**
Average (Single-View)	60.35± 4.85	66.16 ± 6.51	69.84±0.77	66.51± 4.32	68.08± 2.86	**68.28±0.12**

images from the real and synthetic dataset via the cross entropy loss. We compare relative to the Residual-Unet baseline. Results benchmarked in Table 5 and Table 3 reveal that the addition of the Deformable convolutional components has ameliorated segmentation performance over the different integration strategies. This corroborates the ability of the Deformable convolutional component in learning geometric features specific to the dataset at hand. In fact, one can consider a trade-off between the number of Deformable convolutions necessary to increase segmentation performance and the size of the training data. From Table 5 and Table 3, we can gather that the addition of the Deformable component simply at the first stage of the fully Residual U-Net convolutional is sufficient so as to increase segmentation performance significantly. This paves the way to the possibility of learning view-agnostic geometric features via the injection of the Deformable component at simply one encoder/decoder layer within the U-Net like architecture.

Comparing these results relative to the single-view real Fisheye image seg-

Table 4: Mean Intersection over Union (mIoU) results on the WoodScape dataset (Real images) when trained and tested on single-view images, validated over 3 Monte Carlo Simulations

	Residual-Unet	DRU-L(+1)	DRU-Dec(+1)	DRU-EnD(+1)	DRU-EnD(+2)	DRU-EnD(+3)
Road	96.47±0.50	96.28± 0.52	96.36± 0.47	96.14± 0.44	96.65±0.37	96.76± 0.29
Lanemarks	61.65±2.62	62.65± 1.12	57.42± 6.64	61.51± 0.74	61.53±5.07	65.77± 1.70
Curb	73.50±2.68	70.02± 5.39	71.71± 1.76	72.77± 0.47	72.00±2.52	72.11± 0.93
Person	36.12±3.26	29.77±14.94	37.27±11.42	34.88± 3.48	47.44±5.12	56.19±18.96
Rider	28.53±9.12	37.69± 4.92	33.16± 4.80	36.75± 9.96	45.46±6.93	49.19± 4.64
Vehicles	91.41±2.77	91.85± 1.38	91.26± 1.58	90.20± 4.72	93.96±0.26	94.18± 0.30
Bicycle	72.39±5.96	67.67± 1.30	69.28± 4.60	65.71± 6.01	73.14±2.30	67.83± 4.00
Motorcycle	19.35±4.73	18.16± 8.10	21.21± 5.03	20.97±12.76	28.22±3.96	25.45±12.97
Traffic sign	83.93±10.2	88.23± 6.23	83.61±16.50	89.44± 9.30	87.04±8.67	90.58± 9.99
Average	62.59±2.63	62.48± 2.82	62.36± 4.73	63.15± 2.84	67.27±0.67	**68.67± 3.87**

Table 5: Mean Intersection over Union (mIoU) on the WoodScape dataset (Real images) for multi-view, validated over 3 Monte Carlo Simulations

	Residual-Unet	DRU-L(+1)	DRU-Dec(+1)	DRU-EnD(+1)	DRU-EnD(+2)	DRU-EnD(+3)
Road	86.92±2.40	90.45± 2.82	89.70±0.41	91.76± 1.21	90.88±2.32	88.12±2.32
Lanemarks	49.39±6.80	56.43± 6.13	60.15±5.06	65.05± 4.38	62.99±8.57	58.93±5.14
Curb	42.52±3.93	47.67± 3.37	49.18±6.84	56.29± 3.02	52.03±5.35	49.72±4.75
Person	12.99±1.58	22.62± 9.36	22.03±2.33	28.20±10.17	21.02±9.57	19.10±8.34
Rider	6.64±3.44	13.16± 4.19	16.61±5.03	26.25± 5.65	22.87±1.74	18.15±2.27
Vehicles	66.91±0.87	70.27±13.03	72.37±1.26	77.82± 3.83	74.24±7.15	69.53±4.49
Bicycle	13.09±7.26	27.63±10.41	24.40±4.33	31.94± 4.22	24.55±4.21	21.89±2.70
Motorcycle	5.56±4.84	19.05±13.08	16.40±1.80	23.58± 6.19	16.94±5.51	14.33±6.81
Traffic sign	3.50±3.89	8.58± 5.50	7.06±1.20	12.87± 3.88	4.34±1.28	4.38±1.27
Average (Multi-view)	38.08±2.67	45.08± 6.32	45.26±2.64	**50.95± 2.98**	46.50±3.85	43.80±0.79
Average (Front-view)	58.17±5.03	73.91± 7.46	70.82±2.93	**72.16± 4.05**	66.60±8.05	62.19±3.18

mentation experiments in Table 4 and Table 3, we realize that whereas the segmentation performance for front-view segmentation increases as the number of injected Deformable convolutions increases in the ensemble encoder/decoder layers, it is sufficient for multi-view segmentation to add the Deformable component within only the first layer of U-Net like structure. Our intuition to explain the reason is the size of the dataset under consideration. Thus, given small datasets, regular models are rather unable to learn unknown geometric properties due to insufficient samples. Therefore as the number of Deformable components increases, the model is further able to learn the geometric properties better. Given a large dataset, it is only sufficient to add the Deformable component at only one corresponding encoder/decoder layer to guarantee agreeable segmentation performance.

4.4. Data Augmentation Via Surround-view Data

In addition to the multi-view training multi-view testing, we have also registered performances on multi-view training front-view testing. Results are benchmarked in the last row of Table 5 and compared to that of Table 4. Indeed, the training on multi-view images has resulted in an increase in segmentation performance in comparison to training on just single-view data. Thus, whereas the training on single-view data has resulted in a best case performance of about 67 % with the injection of the Deformable components across 3 consecutive layers of the U-Net model, with multi-view training, it was sufficient to just add the Deformable component given only the first layers of the U-Net with a performance that outperformed the single view training by about 72.16 %. In this context, similarities could be drawn between this experiment and the work in [18] that sheds light on the importance of multi-task learning for increasing segmentation performances. Results obtained are promising and pave the way to studying the validity of Deformable convolutions within multi-task networks such as the OmniDet model as proposed by [18]. Moreover, the training on multi-view Fisheye images could be demonstrated as yet another approach for data augmentation via view-agnostic data.

Despite the significance of the proposed method, however, we do admit to certain limitations. By taking a closer look at the tables, we can gather that certain classes are rather small sized objects in the case of terrain (e.g. terrain) or with multiple dis-continuities (e.g. fence). Despite the improved performance over the baseline Residual-Unet, overall segmentation performance is rather limited when it comes to small objects. Thus, the method shares the same limitation with regard to imbalanced-sized objects as the one existing in perspective data. One possible solution is via an augmented memory or memory bank [12] that saves under-presented objects and sample over represented ones in order to ensure a balanced representation of the different classes given the task at hand. Thus, this approach allows the network to have access to more balanced and diverse training data. In addition, maintaining a large dictionary with a queue of data samples as proposed by [12], encourages the model to learn diverse

and representative features. In this way, it can balance the importance given to different instances, including those from underrepresented classes, as they all contribute to building a consistent dictionary. Another possible solution is via the integration of certain prior information regarding the data distribution, characteristics of the classes (size of objects or their location), or domain-specific information to guide the learning process and improve the handling of imbalanced datasets. Similar to [33], the proposed method can distinguish between objects despite varying deformations in their shapes, nevertheless, one could make use of the querying system explored in [33] to enhance the presence and the training for under-represented classes.

To further enhance instance segmentation methods, it is worth exploring training frameworks that address class imbalance, as this remains a challenge in the field. The promising results achieved by the discriminative query embedding learning in boosting query-based models in suggest that incorporating techniques to explicitly handle class imbalance may lead to even more significant performance gains. For instance, considering data augmentation strategies or class re-weighting during training could help the model better cope with underrepresented classes, ultimately improving segmentation accuracy. Additionally, investigating methods that dynamically adjust the importance given to different instances based on their rarity or difficulty may also contribute to better handling class imbalance in the context of instance segmentation. Overall, by integrating these considerations into the training process, we anticipate further progress in developing robust and accurate instance segmentation models for real-world applications. Integrating Deformable convolution within SgNet could also be an option for handling class imbalance. Sg-Nets are networks that can handle class imbalance via dynamic mask predictions of sub-regions within videos/images and via multi-task learning (detection, segmentation, and tracking). By allowing the tasks to share features and co-adapt, it could potentially improve the overall performance on all classes, including the less frequent ones [20].

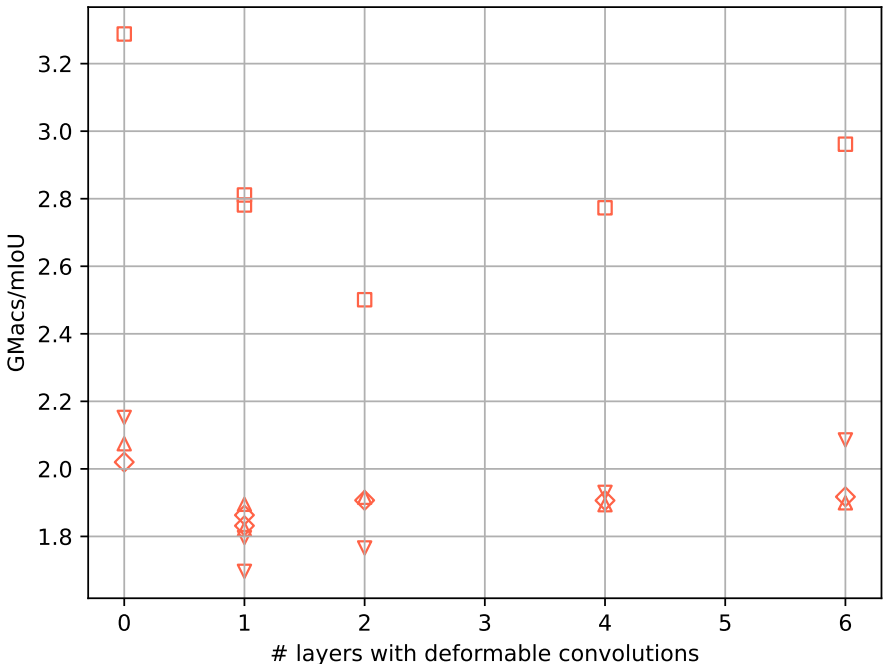


Figure 8: Segmentation performance versus workload of the different models:

□: WoodScape/Multiple views ▽: WoodScape/Single view

◇: SynWoodScape/Multiple views △: SynWoodScape/Single view

4.5. Models efficiency

We define the efficiency of the model as the ratio of the workload (GMACs) over the IoU. Figure 8 presents the results for the four models: single view versus multiple views, WoodScape versus SynWoodScape. The baseline model without deformable layers is systematically requiring more GMACs per IoU. The best efficient tends is obtained when either a single layer from the encoder or the decoder applied a Deformable convolution. The gap tends to narrow as more Deformable convolutional layers are added to the network layers. Still, the fully Deformable residual-Unet (6 layers) outperforms the baseline. As a result, the Deformable layers induce a more efficient network for fully residual-UNets. The behavior for SynWoodScape is similar for single and multiple views. The close IoU values of these two models on SynWoodScape explains this result. For the same reasons the gap is rather broader for WoodScape.

We would like to note here that increasing the number of Deformable convolution injections into the Unet architecture may not mean an increase in segmentation performance over the architectures with lower injections. The lack of increasing improvement when incorporating additional deformable convolutions could be related to the possibility that the deformable convolutions may already capture the necessary spatial information within the existing layers of the model. As a result, introducing more Deformable convolutions may not significantly contribute to the overall performance as the model's architecture carried with the Deformable convolution may have reached a performance saturation. To achieve further advancements in segmentation performance, more intricate architectures with greater capacity for capturing the Deformable characteristics of objects would be necessary.

5. Conclusion and Future Work

In this paper, we have investigated Deformable convolutions for Fisheye image segmentation. The proposed method shows the promising potential of Deformable convolutions in modeling unknown geometric transformations and

distortions existent in Fisheye images. Using the proposed method we achieved an investigation study with multiple integration strategies of the deformable convolutional blocks in the residual Unet. We further shed light on the importance of exploiting surround-view data as an effective data augmentation method for front-view Fisheye image segmentation. The experiments have shown for the front view that the more we add deformable convolutional blocks the more the results improve. We also found out that training on surround view images improves the results on the front view compared with when we just train on the front view images. Finally, we highlights the increased efficiency of fully residual U-Net when Deformable convolutions substitutes rectangular convolutions on both WoodScape and SynWoodScape datasets.

The conducted experiments have shown that the deformable convolutional blocks offer a finer and more efficient modelling for Fisheye images, therefore future work may involve the integration of these blocks into other backbone architecture or multi task networks dedicated for omnidirectional images or for other tasks like instance segmentation, detection and optical flow estimation. It may also involve integration of prior knowledge regarding the objects possible positions, and shapes, or depth maps, as constraints in order to improve segmentation performances and counter-react the class size imbalance within the datasets.

In recent years, transformer-based approaches have emerged as powerful tools for image segmentation [34, 19]. These approaches leverage the strengths of transformers in capturing long-range dependencies and context information that may potentially make it a powerful tool for data with larger fields of view. While such approaches have shown promising results in semantic segmentation tasks, its applicability to Fisheye data remains unexplored. In this context, an intriguing avenue for future research lies in exploring the potential synergy between transformer-based approaches and Deformable convolutions. Fisheye images pose unique challenges due to their distorted perspective and wide field of view. By integrating Deformable convolutions into the Transformer framework for segmentation, it may be possible to enhance the model's ability

to capture and adapt to the deformations and irregularities inherent in fisheye images while preserving the transformer model's computational efficiency. For example, the authors of [33] propose a new training framework to improve query-based instance segmentation methods, which also fall under the category of image segmentation. The framework leverages dataset-level uniqueness and transformation equivariance to enhance instance separation and achieve more robust instance-query matching. By querying instances across the entire training dataset, the model learns more discriminative queries, resulting in significant performance gains on benchmark datasets. Such architecture and method could benefit from the previliges of Deformable convolutions in order to learn complex patterns in the data.

Interpretability methods aim to provide insights into how a neural network makes decisions and quantifies the importance of different components or operations within the network architecture. In this context, future work could also include exploring interpretability tools and techniques [32, 28] in order to quantify and asses the true added value of deformable convolutions and how they impact networks performance and behavior.

Adversarial attacks are known to exploit the weaknesses of deep learning models with semantic segmentation models being no exception to the rule [5]. Despite the ability of the proposed model to learn unknown geometric distortions from Fisheye images, future work may include studying further the robustness of the model relative to adversarial distortions, which are carefully crafted to deceive the segmentation network. Furthermore, there is an opportunity to gain insights from alternative models on their approach to handling adversarial attacks and incorporate these techniques into our own network [5]. Finally, it is crucial to assess the model's capacity for resilience and robustness when confronted with perturbed inputs, especially under adversarial conditions.

Acknowledgments

The authors would like to acknowledge the ANR (Project APi, grant ANR-18-CE23-0014) for funding this project and the CRIANN for providing computational resources. This work was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19- P3IA-0001 (PRAIRIE 3IA Institute) and ANR-10-IAIHU-06.

Declarations

Competing interests.

The authors declare that there is no conflict of interest regarding the publication of this manuscript.

Data Availability

The datasets used during the current study are available in https://drive.google.com/drive/folders/1X5JOMEfVlaXfdNy24P8VA-jMs0yzf_HR

Code availability

The code is available on Github via the following link: https://github.com/rosanajurdi/WoodScape-Segmentation-Project

Authors' contributions

. All authors contributed to the study conception setup, analysis, and proofreading.

Ethics approval.

Approved

Consent to participate

Approved

Consent for publication

Approved

References

[1] Ahmad, O., Lecue, F., 2022. Fisheyehdk: Hyperbolic deformable kernel learning for ultra-wide field-of-view image recognition. Proceedings of the AAAI Conference on Artificial Intelligence 36, 5968–5975. URL: https://ojs.aaai.org/index.php/AAAI/article/view/20542, doi:10.1609/aaai.v36i6.20542.

[2] Blott, G., Takami, M., Heipke, C., 2019. Semantic Segmentation of Fisheye Images, in: Leal-Taix?, L., Roth, S. (Eds.), Computer Vision ? ECCV 2018 Workshops. Springer International Publishing, Cham. volume 11129, pp. 181–196. doi:10.1007/978-3-030-11009-3_10. series Title: Lecture Notes in Computer Science.

[3] Brostow, G.J., Fauqueur, J., Cipolla, R., 2009. Semantic object classes in video: A high-definition ground truth database. Pattern Recognition Letters 30, 88–97. URL: https://www.sciencedirect.com/science/article/pii/S0167865508001220, doi:https://doi.org/10.1016/j.patrec.2008.04.005.

[4] Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R., 2008. Segmentation and recognition using structure from motion point clouds, in: ECCV (1), pp. 44–57.

[5] Cheng, Z., Liang, J., Choi, H., Tao, G., Cao, Z., Liu, D., Zhang, X., 2022. Physical attack on monocular depth estimation with optimal adversarial patches, in: ECCV (38), pp. 514–532. URL: https://doi.org/10.1007/978-3-031-19839-7_30.

[6] Cohen, T., Welling, M., 2016. Group equivariant convolutional networks, in: Balcan, M.F., Weinberger, K.Q. (Eds.), Proceedings of The 33rd Inter-

national Conference on Machine Learning, PMLR, New York, New York, USA. pp. 2990–2999.

[7] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B., 2016. The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[8] Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y., 2017. Deformable convolutional networks, in: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 764–773. doi:10.1109/ICCV.2017.89.

[9] Deng, L., Yang, M., Li, H., Li, T., Hu, B., Wang, C., 2019. Restricted deformable convolution-based road scene semantic segmentation using surround view cameras. IEEE Transactions on Intelligent Transportation Systems 21, 4350–4362.

[10] Deng, L., Yang, M., Qian, Y., Wang, C., Wang, B., 2017. Cnn based semantic segmentation for urban traffic scenes using fisheye camera, in: 2017 IEEE Intelligent Vehicles Symposium (IV), pp. 231–236. doi:10.1109/IVS.2017.7995725.

[11] Eder, M., Shvets, M., Lim, J., Frahm, J.M., 2020. Tangent images for mitigating spherical distortion, in: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12426–12434.

[12] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020. Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

[13] Hu, X., An, Y., Shao, C., Hu, H., 2022. Distortion convolution module for semantic segmentation of panoramic images based on the image-forming principle. IEEE Transactions on Instrumentation and Measurement 71, 1–12. doi:10.1109/TIM.2021.3139710.

[14] Huang, Y., Chen, Y., 2020. Survey of state-of-art autonomous driving technologies with deep learning, in: 2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 221–228. doi:10.1109/QRS-C51114.2020.00045.

[15] Jeon, Y., Kim, J., 2017. Active convolution: Learning the shape of convolution for image classification, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA. pp. 1846–1854. doi:10.1109/CVPR.2017.200.

[16] Jiang, C.M., Huang, J., Kashinath, K., Prabhat, Marcus, P., Niessner, M., 2019. Spherical CNNs on unstructured grids, in: International Conference on Learning Representations.

[17] Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., Ben Ayed, I., 2019. Boundary loss for highly unbalanced segmentation, in: Medical Imaging with Deep Learning, PMLR, London, UK. pp. 285–296.

[18] Kumar, V.R., Yogamani, S.K., Rashed, H., Sistu, G., Witt, C., Leang, I., Milz, S., Mäder, P., 2021. Omnidet: Surround view cameras based multi-task visual perception network for autonomous driving. CoRR abs/2102.07448. arXiv:2102.07448.

[19] Liang, J., Zhou, T., Liu, D., Wang, W., 2023. Clustseg: Clustering for universal segmentation. arXiv:2305.02187.

[20] Liu, D., Cui, Y., Tan, W., Chen, Y., 2021. Sg-net: Spatial granularity network for one-stage video instance segmentation. arXiv:2103.10284.

[21] Playout, C., Ahmad, O., Lecue, F., Cheriet, F., 2021a. Adaptable deformable convolutions for semantic segmentation of fisheye images in autonomous driving systems. arXiv preprint arXiv:2102.10191 .

[22] Playout, C., Ahmad, O., Lécué, F., Cheriet, F., 2021b. Adaptable deformable convolutions for semantic segmentation of fisheye images in au-

tonomous driving systems. CoRR abs/2102.10191. URL: https://arxiv.org/abs/2102.10191, arXiv:2102.10191.

[23] Quan, T.M., Hildebrand, D.G.C., Jeong, W.K., 2021. Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics. Frontiers in Computer Science 3. doi:10.3389/fcomp.2021.613981.

[24] Romera, E., Álvarez, J.M., Bergasa, L.M., Arroyo, R., 2018. ERFNet: Efficient residual factorized convnet for real-time semantic segmentation. IEEE Transactions on Intelligent Transportation Systems 19, 263–272.

[25] Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: MICCAI, pp. 234–241.

[26] Sáez, Á., Bergasa, L., López-Guillén, E., Romera, E., Tradacete, M., Gómez-Huélamo, C., del Egido, J., 2019. Real-time semantic segmentation for fisheye urban driving images based on ERFNet. Sensors 19, 503. doi:10.3390/s19030503.

[27] Saez, A., Bergasa, L.M., Romeral, E., Lopez, E., Barea, R., Sanz, R., 2018. CNN-based fisheye image real-time semantic segmentation, in: 2018 IEEE Intelligent Vehicles Symposium (IV), IEEE. pp. 1039–1044. doi:10.1109/ivs.2018.8500456.

[28] Salahuddin, Z., Woodruff, H.C., Chatterjee, A., Lambin, P., 2022. Transparency of deep neural networks for medical image analysis: A review of interpretability methods. Computers in Biology and Medicine 140, 105111. URL: https://www.sciencedirect.com/science/article/pii/S0010482521009057, doi:https://doi.org/10.1016/j.compbiomed.2021.105111.

[29] Sekkat, A.R., Dupuis, Y., Honeine, P., Vasseur, P., 2022a. A comparative study of semantic segmentation of omnidirectional images from

a motorcycle perspective. Scientific Reports 12, 4968. doi:10.1038/s41598-022-08466-9.

[30] Sekkat, A.R., Dupuis, Y., Kumar, V.R., Rashed, H., Yogamani, S., Vasseur, P., Honeine, P., 2022b. Synwoodscape: Synthetic surround-view fisheye camera dataset for autonomous driving. IEEE Robotics and Automation Letters 7, 8502–8509.

[31] Sekkat, A.R., Dupuis, Y., Vasseur, P., Honeine, P., 2020. The omniscape dataset, in: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 1603–1608. doi:10.1109/ICRA40945.2020.9197144.

[32] Wang, W., Han, C., Zhou, T., Liu, D., 2023. Visual recognition with deep nearest centroids, in: The Eleventh International Conference on Learning Representations. URL: https://openreview.net/forum?id=CsKwavjr7A.

[33] Wang, W., Liang, J.C., Liu, D., 2022. Learning equivariant segmentation with instance-unique querying, in: Oh, A.H., Agarwal, A., Belgrave, D., Cho, K. (Eds.), Advances in Neural Information Processing Systems. URL: https://openreview.net/forum?id=q0XxMcbaZH9.

[34] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P., 2021. Segformer: Simple and efficient design for semantic segmentation with transformers, in: Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (Eds.), Advances in Neural Information Processing Systems. URL: https://openreview.net/forum?id=OG18MI5TRL.

[35] Yin, X., Wang, X., Yu, J., Zhang, M., Fua, P., Tao, D., 2018. Fisheyerecnet: A multi-context collaborative deep network for fisheye image rectification, in: Proceedings of the European Conference on Computer Vision (ECCV).

[36] Yogamani, S., Hughes, C., Horgan, J., Sistu, G., Varley, P., O'Dea, D., Uricar, M., Milz, S., Simon, M., Amende, K., Witt, C., Rashed, H., Chennupati, S., Nayak, S., Mansoor, S., Perrotton, X., Perez, P., 2019. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving,

in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).