FLORE
Repository istituzionale dell'Università degli Studi di Firenze

FLODCAST: Flow and depth forecasting via multimodal recurrent architectures

Questa è la Versione finale referata (Post print/Accepted manuscript) della seguente pubblicazione:

Original Citation:

FLODCAST: Flow and depth forecasting via multimodal recurrent architectures / Ciamarra, A., Becattini, F., Seidenari, L., Del Bimbo, A.. - In: PATTERN RECOGNITION. - ISSN 0031-3203. - ELETTRONICO. - (2024), pp. 0-0. [10.1016/j.patcog.2024.110337]

Availability:

The webpage https://hdl.handle.net/2158/1350022 of the repository was last updated on 2024-02-13T15:48:30Z

Published version:

DOI: 10.1016/j.patcog.2024.110337

Terms of use:

Open Access

La pubblicazione è resa disponibile sotto le norme e i termini della licenza di deposito, secondo quanto stabilito dalla Policy per l'accesso aperto dell'Università degli Studi di Firenze (https://www.sba.unifi.it/upload/policy-oa-2016-1.pdf)

Publisher copyright claim:

La data sopra indicata si riferisce all'ultimo aggiornamento della scheda del Repository FloRe - The above-mentioned date refers to the last update of the record in the Institutional Repository FloRe

(Article begins on next page)

05 September 2026

Pattern Recognition 150 (2024) 110337

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

FLODCAST: Flow and depth forecasting via multimodal recurrent architectures

Andrea Ciamarra [a], Federico Becattini [b,*], Lorenzo Seidenari [a], Alberto Del Bimbo [a]

[a] Dipartimento di Ingegneria dell'Informazione, University of Florence, Italy
[b] Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche, University of Siena, Italy

ARTICLE INFO

Keywords:
Depth forecasting
Optical flow forecasting
Segmentation

ABSTRACT

Forecasting motion and spatial positions of objects is of fundamental importance, especially in safety-critical settings such as autonomous driving. In this work, we address the issue by forecasting two different modalities that carry complementary information, namely optical flow and depth. To this end we propose FLODCAST a flow and depth forecasting model that leverages a multitask recurrent architecture, trained to jointly forecast both modalities at once. We stress the importance of training using flows and depth maps together, demonstrating that both tasks improve when the model is informed of the other modality. We train the proposed model to also perform predictions for several timesteps in the future. This provides better supervision and leads to more precise predictions, retaining the capability of the model to yield outputs autoregressively for any future time horizon. We test our model on the challenging Cityscapes dataset, obtaining state of the art results for both flow and depth forecasting. Thanks to the high quality of the generated flows, we also report benefits on the downstream task of segmentation forecasting, injecting our predictions in a flow-based mask-warping framework.

1. Introduction

Forecasting capabilities are fundamental for autonomous driving systems. Being able to predict possible hazards in the surrounding environment allows the vehicle to plan ahead and improve decision-making for safe navigation. The problem can be addressed from many angles, e.g. processing the motion of individual agents or jointly reasoning about the evolution of the whole environment with respect to the driver. In this paper, we frame the future prediction problem as the task of forecasting both optical flow and depth maps in a joint multi-task, multi-modal framework called FLODCAST (FLOw and Depth foreCASTing). Despite several works addressing the problem by forecasting individual trajectories [1], reasoning about social behaviors [2] or predicting where agents will appear in the scene [3–6], forecasting holistic information about the environment has shown a few promising results in the literature [7–9]. These works tend to forecast different modalities such as optical flow [9,10], depth [11,12] or semantic segmentation [7,8,13] to allow reasoning about the whole environment, including moving agents and static environment as well. This entails that a learning method addressing such a task must take into account several challenging factors including ego-motion, environmental cues and apparent motion of the agents from the vehicle's perspective. We choose to combine optical flow and depth as they provide insights about complementary information: on the one hand, optical flow encodes the motion of others as well as the ego-speed; on the other hand, depth enables to contextualize such motion in a 3D space. Furthermore, our results show that a joint processing of the two modalities improves the forecasting capabilities in both domains compared to single-modality reasoning, thanks to information sharing across modalities.

In summary, instead of casting the problem from a high-level perspective as done in most of prior work, we choose to address the problem from a lower level. We forecast finer-grained information such as pixel-level optical flows and depth maps, which provide a more comprehensive source of information. Nonetheless, we show that access to such information can then be leveraged to infer high-level aspects such as forecasting semantic instances.

The design of prior work anticipating low level information such as flow or depth present certain limitations. Some methods [9,11,14] focus solely on training to predict the next frame, applying the model autoregressively to extend predictions into the future. However, this autoregressive approach is susceptible to error accumulation over time. In contrast, other approaches [10,12] eliminate intermediate predictions

* Corresponding author.
E-mail addresses: andrea.ciamarra@unifi.it (A. Ciamarra), federico.becattini@unisi.it (F. Becattini), lorenzo.seidenari@unifi.it (L. Seidenari), alberto.delbimbo@unifi.it (A. Del Bimbo).

https://doi.org/10.1016/j.patcog.2024.110337
Received 7 September 2023; Received in revised form 26 January 2024; Accepted 9 February 2024
Available online 12 February 2024
0031-3203/© 2024 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

and instead train models to predict information at a specific time step in the future. Unfortunately, these latter approaches require training models for specific future frames and are unable to provide continuous estimates over time. We address these limitations by directly forecasting multiple time steps at a time, yet maintaining the model autoregressive to avoid the need for training timestep-specific models. We find that training with long-term supervision leads to smaller errors at inference time.

Summarizing, our main contributions are the following:

(i) We propose FLODCAST, a FLOw and Depth foreCASTing network that jointly estimates optical flow and depth for future frames autoregressively, showing that information sharing across tasks proves highly beneficial for forecasting.
(ii) FLODCAST architecture mitigates the accumulation of errors that typically impede the performance of single-step autoregressive models and eliminates the need for different models with different prediction horizons.
(iii) We show that our joint estimate improve both modalities yielding state-of-the-art results which also reflect on downstream tasks such as instance segmentation forecasting.

2. Related work

Depth forecasting. Several works have focused on learning to infer depth from monocular RGB cameras [15–17]. Nonetheless, relying on depth estimators on predicted future RGBs is hard, due to high uncertainty in predicting raw pixels [18–22]. Therefore, other works propose to deal with depth anticipation for future frames, mostly known in the literature as depth forecasting or video depth forecasting. Qi et al. [14] introduce an entire framework for predicting 3D motion (both optical flow and depth map) and synthesizing the RGB with its semantic map for unobserved future frames. To this end, they leverage images, depth maps and semantic segmentations of past frames but they make predictions limited to the subsequent future frame, i.e. at the frame t+1. Also limited to a single future timestep, Hu et al. [11] design a probabilistic model for future video prediction, where scene features are learned from input images and are then used to build spatio-temporal representations, incorporating both local and global contexts. These features are finally fed into a recurrent model with separate decoders, each one forecasting semantic segmentation, depth and dense flow at the next future frame. Nag et al. [12] propose a self-supervised method for depth estimation directly at the kth frame after the last observed one, i.e. at $t+k$. By means of a feature forecasting module, they learn to map pyramid features extracted from past sequences of both RGBs and optical flows to future features, exploiting a series of ConvGRUs and ConvLSTMs for spatio-temporal relationships in the past. With the same goal, Boulahbal et al. [23] design an end-to-end self-supervised approach by using a hybrid model based on CNN and Transformer that predicts depth map and ego-motion at $t+k$ by processing an input sequence of past frames. Differently from prior work, we predict both dense optical flows and depth maps, also leveraging both modalities as inputs. We directly predict several timesteps ahead simultaneously while retaining autoregressive capabilities, that allows the model to accurately predict far into the future.

Flow forecasting. Optical flow estimation has been largely studied in the past [24,25]. Consolidated deep learning approaches have addressed this problem with promising results [26–28], also exploiting transformer-based architectures [29–31]. However, these methods are designed to estimate the optical flow by accessing adjacent frames as they are available to the network. Different approaches have been introduced incorporating optical flow features to infer imminent future scenarios under different points of view, such as predicting depth maps [12], semantic segmentations [8,13] and instance segmentations [9]. Multitasking methods also exist [10,14,32].

Many works leverage motion features for future predictions to perform several specific tasks, ranging from semantic segmentation [7,8,10,13], instance-level segmentation [9] and depth estimation [11,12,14]. However, just a few approaches have specifically addressed the task of optical flow forecasting, i.e. the problem of anticipating the optical flow for future scenes. Jin et al. [10] was the first to propose a framework, which jointly predicted optical flow and semantic segmentation for the next frame using the past ones. To make predictions for multiple time steps, they just iterate a two-step finetuned model so to alleviate the propagation error. Ciamarra et al. [9] instead introduced OFNet, a recurrent model able to predict the optical flow for the next time step exploiting spatio-temporal features from a ConvLSTM. Such features are learned to generate a sequence of optical flows shifted by one time step ahead from the input sequence. Without finetuning, the recurrent nature of the model allows OFNet to make predictions for any time steps ahead. Considering the high uncertainty of the future, all the proposed methods [9,10,13,14,32] are typically trained to make predictions at the single time step ahead, and then used for the future ones by autoregressively providing in input the predictions obtained at the previous iterations. We, instead, address a more general forecasting task, with the purpose of providing future optical flows directly for multiple time steps ahead, by exploiting both past flows and the corresponding depth maps. We also make use of depth maps as input because our framework is designed as a novel multitask and multimodal approach to also generate future depth maps.

To the best of our knowledge, we are the first to jointly forecast optical flows and depth maps for multiple consecutive frames into the future. Besides, we do not require other information (even during training), like camera pose estimation, which is usually needed to deal with monocular depth estimation.

3. Method

In this work we introduce FLODCAST, a novel approach for predicting optical flow and depth map jointly for future unobserved frames from an ego-vehicle perspective applied to autonomous driving context.

3.1. Problem definition

Given a sequence $\mathbf{S} = \{I_t\}$ of frames, let $\mathbf{D} = \{D_1, D_2, \dots, D_T\}$ be the depth map sequence extracted from the last T frames of $\mathbf{S}$. Likewise, we define $\mathbf{OF} = \{OF_1, OF_2, \dots, OF_T\}$ the corresponding optical flows computed every two consecutive frames in $\mathbf{S}$, such that $OF_t = Flow(I_{t-1}, I_t)$, with $t \in [1, T]$, encodes the motion of the source frame I_{t-1} onto the target frame I_t. Our purpose is to anticipate flow and depth maps for future frames after K time instants, i.e. forecasting D_{T+K} and OF_{T+k} for the frame I_{T+K}.

The importance of jointly anticipating flow and depth stems from the nature of the two modalities. Optical flow is a two-dimensional projection of the three-dimensional motion of the world onto the image plane [33]. An object in the foreground moving fast produces a large displacement, whereas when it comes far from the observer, moving at the same speed, it generates a very small displacement. Therefore, knowledge about the depth of such an object can help to model its future dynamics. Vice-versa, observing the motion of an object can provide information about its distance from the camera. Overall, by jointly modeling optical flow and depth we can represent the 3D scene displacement at time t in terms of the components (u, v, d, t), where (u, v) are the horizontal and vertical components of OF_t and d is the depth map.

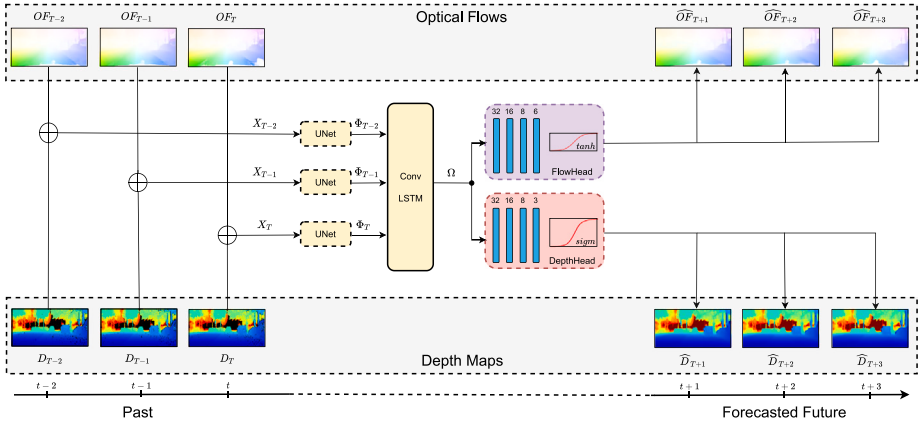

Fig. 1. FLODCAST forecasts both future flows and depth maps from the past ones autoregressively. For each time step, we aggregate flow and depth at the last channel (by the concatenation operator, $\oplus$), then 64-channel features are extracted through a UNet [34] backbone. Finally, predictions are obtained from two dedicated fully convolutional heads.

3.2. Flow and depth forecasting via multimodal recurrent architectures

We design **FLODCAST**, a novel optical **FLO**w and **D**epth fore**CAST**ing network that anticipates both modalities at each future time step by observing the past ones. An overview of FLODCAST is shown in Fig. 1.

FLODCAST takes a sequence $X = \{X_1, X_2, \dots, X_T\}$ of T past observations composed of dense optical flows and depth maps. In detail, each X_t encodes the input features for the image I_t in the past, that are obtained by concatenating the optical flow OF_t with the depth map D_t. In other words, $X_t = (OF_t \oplus D_t)$. We use a shared UNet to compute an intermediate representation Φ_t for each X_t. $X_{T-K}, X_{T-K+1}, \dots, X_T$ are then forwarded into our ConvLSTM module to extract our future prediction feature Ω that is used as an input for the two final branches. The model generates as output a sequence $\hat{X} = \{\hat{X}_{T+1}, \hat{X}_{T+2}, \dots, \hat{X}_{T+K}\}$, that is a sequence of K future optical flows and K depth maps. We set $T = 3$ and $K = 3$ in all our experiments.

Since optical flows and depth maps encode very different information about the scene, we add two separate heads after extracting features from the input in order to handle multimodal predictions. Therefore, we feed in input a sequence of concatenated optical flows and depths $\{X_1, X_2, \dots, X_T\}$ to a single recurrent ConvLSTM network, in which a UNet backbone is used to extract features at 64 channels for each input X_t, $t = 1, \dots, T$, so to output a tensor of size $(H \times W \times 64)$, where $(H \times W)$ is the input resolution. Our feature extractor is the same UNet architecture as in [9], i.e. a fully convolutional encoder–decoder network with skip connections, consisting of 5 layers with filters $\{64, 128, 256, 512, 1024\}$ respectively. These 64-channel features capture meaningful spatio-temporal contexts of the input representation. The features are then passed to the two convolutional heads, which are end-to-end trained to simultaneously generate the sequence of future optical flows and depth maps (respectively depicted by the purple and the red blocks in the right side of Fig. 1). Each head is a fully convolutional network made of sequences of Conv2D+ReLUs with $\{32, 16, 8\}$ filters. Finally, we append at the end of the optical flow head a convolution operation with $2 \times K$ channels and we use a *tanh* activation function, so to produce the (u, v) flow field values normalized in $(-1, 1)$. Instead, after the depth head, we attach a convolution operation with a K channels and a sigmoid activation in order to get depth maps normalized in $(0, 1)$. Instead of outputting one prediction at a time as in prior work [9], we directly generate K flows and depth maps simultaneously, to make the model faster compared to autoregressive models which would require looping over future steps.

3.3. Loss

To train FLODCAST we compute a linear transformation of the original input values, by rescaling depth map values in $[0, 1]$ and optical flows in $[-1, 1]$ through a min–max normalization, with minimum and maximum values computed over the training set. Inspired by [35], we use the reverse Huber loss, called *BerHu* for two main reasons: (i) it has a good balance between the two L1 and L2 norms since it puts high weight towards values with a high residual, while being sensitive for small errors; (ii) it is also proved to be more appropriate in case of heavy-tailed distributions [35], that perfectly suits our depth distribution, as shown in Fig. 2. BerHu minimizes the prediction error, through either the L2 or L1 loss according to a specific threshold c calculated for each batch during the training stage. Let $x = \hat{y} - y$ be the difference between the prediction and the corresponding ground truth. This loss $\mathcal{B}(x)$ is formally defined as:

$$\mathcal{B}(x) = \begin{cases} |x|, & |x| \le |c| \\ \frac{x^2 + c^2}{2c}, & \text{otherwise} \end{cases} \tag{1}$$

Thus, we formulate our compound loss, using a linear combination of the optical flow loss $\mathcal{L}_{\text{flow}}$ and the depth loss $\mathcal{L}_{\text{depth}}$ (Eq. (2)):

$$\mathcal{L} = \alpha\, \mathcal{L}_{\text{flow}} + \beta\, \mathcal{L}_{\text{depth}} \tag{2}$$

Specifically, we apply the reverse Huber loss to minimize both the optical flow and depth predictions, using the same loss formulation, since the threshold c is computed for each modality, and that value depends on the current batch data. Therefore, $\mathcal{L}_{\text{flow}}$ is the loss function for the optical flow computed as:

$$\mathcal{L}_{\text{flow}} = \frac{1}{M} \sum_{j=1}^{M} \mathcal{B}(|OF_j - \widehat{OF}_j|) \tag{3}$$

where $M = B \times R \times 2$, since the flow field has (u, v) components over R image pixels and B is the batch size, whereas OF_j and $\widehat{OF}_j$ are the optical flows, respectively of the ground truth and the prediction at the pixel j. Likewise, we do the same for the depth loss $\mathcal{L}_{\text{depth}}$:

$$\mathcal{L}_{\text{depth}} = \frac{1}{P} \sum_{j=1}^{P} \mathcal{B}(|D_j - \hat{D}_j|) \tag{4}$$

where $P = B \times R$, D_j and $\hat{D}_j$ are the depth maps, respectively of the ground truth and the prediction at the pixel j. We follow [35] and we set $c = \frac{1}{5} max_j(|y_j - \hat{y}_j|)$, i.e. the 20% of the maximum absolute error between predictions and ground truth in the current batch over all pixels.

4. Results

In this section we report our forecasting results on Cityscapes [36] for the depth and flow forecasting tasks. We first describe the experimental setting and the metrics used to evaluate our approach.

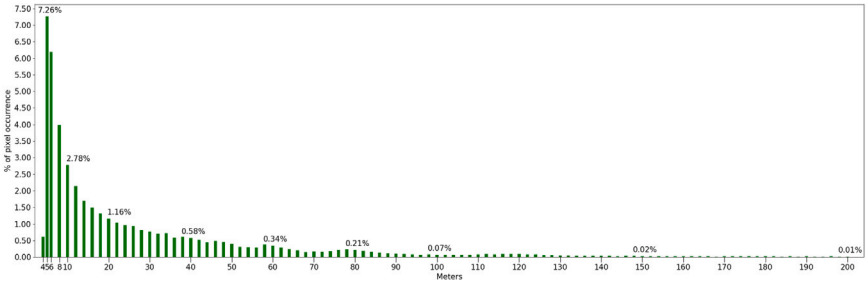

Fig. 2. Distribution of depth values grouped by distance on the Cityscapes training set. Note that depth values below 3 m are not present in the dataset.

Then, we present our results, comparing FLODCAST to state-of-the-art approaches. We also present ablation studies to better highlight the importance of all the modules in the architecture and some failure cases. Besides, in Section 5, we show that our approach can be easily applied to downstream tasks such as semantic segmentation and instance segmentation forecasting, demonstrating improvements, especially at farther prediction horizons.

4.1. Dataset

For evaluation, we use Cityscapes [36], which is a large urban dataset with very challenging dynamics, recorded in several German cities. Each sequence consists of 30 frames at a resolution of 1024×2048. Cityscapes contains 5000 sequences, split in 2975 for train, 500 for validation and 1525 for testing. Different annotations are available. In particular, we leverage precomputed disparity maps for all frames, from which depth maps can be extracted through the camera parameters. There are also both instance and semantic segmentations that are available at the 20-th frame of each sequence.

4.2. Experimental setting

We compute optical flows using FLowNet2 [26] (pretrained FlowNet2-c) and rescale them according to the maximum and minimum values in the training set, so to have normalized values in $(-1, 1)$. Depth maps D are obtained using disparity data d and camera parameters (focal length f and baseline b), i.e. by computing $D = f \cdot b/d$. Invalid measurements or zero-disparity values are set to 0. To normalize depth maps, we observe that most depth values fall within 150 m in the training set (Fig. 2). Thus, we cap values at 150 m and then normalize them in $(0, 1)$. All frames are rescaled at 128×256 px for both data sources to accelerate learning. We train FLODCAST for 30 epochs using Adam and learning rate 0.0001. To balance the two losses in Eq. (2), we set $\alpha = 10$ and $\beta = 1$. At inference time we recursively employ the model by feeding as input previous predictions to reach farther time horizons. We provide outputs at a resolution of 256×512, following [37], by doubling the resolution. FLODCAST has approximately 31.4M trainable parameters. The whole training takes 58 h on a single GPU NVIDIA Titan RTX with 24 GB using a batch size of 12.

4.3. Evaluation metrics

We quantitatively evaluate depth forecasting using standard metrics as in [38]: (i) absolute relative difference (AbsRel), (ii) squared relative difference (SqRel), (iii) root mean squared error (RMSE) and (iv) logarithmic scale-invariant RMSE (RMSE-Log), defined as follows:

$$\text{AbsRel} = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{y_i} \tag{5}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|^2} \tag{6}$$

$$\text{SqRel} = \frac{1}{N}\sum_{i=1}^{N}\frac{(y_i - \hat{y}_i)^2}{y_i} \tag{7}$$

$$\text{RMSE-Log} = \frac{1}{N}\sum_{i=1}^{N} d_i^2 - \frac{1}{N^2}\left(\sum_{i=1}^{N} d_i\right)^2 \tag{8}$$

where y and $\hat{y}$ are the ground truth and the prediction, each with N pixels indexed by i, while $d = \log \hat{y} - \log y$ is their difference in logarithmic scale. AbsRel and SqRel are errors that can be also calculated at pixel-level, instead RMSE, RMSE-Log measure mistakes averaged on the whole image. In particular, AbsRel draws attention to the absolute difference between the prediction and the target with respect to the ground truth itself (e.g. an AbsRel of 0.1 means that the error is 10% of the ground truth), which makes it suitable for a fine-grained understanding. The SqRel instead emphasizes large errors since the difference is squared. RMSE is the root of the mean squared errors while RMSE-Log, introduced in [38], is an L2 loss with a negative term used to keep relative depth relations between all image pixels, i.e an imperfect prediction will have lower error when its mistakes are consistent with one another.

We also measure the percentage of inliers with different thresholds [38], i.e. the percentage of predicted values $\hat{y}_i$ for which the ratio δ with the ground truth y_i is lower than a threshold τ:

$$\%\text{ of } \hat{y} \text{ s.t. } \max\left(\frac{y_i}{\hat{y}_i}, \frac{\hat{y}_i}{y_i}\right) = \delta < \tau \tag{9}$$

with $\tau = \{1.25, 1.25^2, 1.25^3\}$.

We assess the performance of the flow forecasting task, by computing the mean squared error between the prediction and the groundtruth on both the two flow channels, using Eq. (10), and averaging them, as done in [9]:

$$\text{MSE}_c = \frac{1}{H\,W}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(f_c(i,j) - \hat{f}_c(i,j)\right)^2 \tag{10}$$

where MSE_c is the error referred to the channel $c := \{u, v\}$ between the ground truth optical flow field $f_c(i,j)$ and the prediction $\hat{f}_c(i,j)$ at the pixel (i,j) and H and W is height and width respectively. We also report the average end-point-error EPE [39], which measures the per-pixel euclidean distance between the prediction and the ground truth averaged among all the image pixels:

$$\text{EPE} = \frac{1}{H\,W}\sum_{i=1}^{H\,W}\sqrt{(\hat{u}_i - u_i)^2 + (\hat{v}_i - v_i)^2} \tag{11}$$

where (u_i, v_i) are the horizontal and vertical components of the optical flow ground truth, likewise $(\hat{u}_i, \hat{v}_i)$ are the corresponding components of the prediction, at the ith pixel.

4.4. Future depth estimation

We evaluate our approach for future depth estimation on Cityscapes. As in prior works, e.g. [11], we evaluate our method after $t + k$ frames,

Table 1
Quantitative results for depth forecasting after $t+k$ on Cityscapes test set, both at short-term and mid-term predictions, i.e. at $k = 5$ and $k = 10$ respectively.

Short term $k = 5$							
	Lower is better ↓				Higher is better ↑		
Method	AbsRel	SqRel	RMSE	RMSE-Log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Copy last	0.257	4.238	7.273	0.448	0.765	0.893	0.940
Qi et al. [14]	0.208	1.768	6.865	0.283	0.678	0.885	0.957
Hu et al. [11]	0.182	1.481	6.501	0.267	0.725	0.906	0.963
Sun et al. [4]	0.227	3.800	6.910	0.414	0.801	0.913	0.950
Goddard et al. [41]	0.193	1.438	5.887	0.234	0.836	0.930	0.958
DeFNet [12]	0.174	1.296	5.857	0.233	0.793	0.931	0.973
FLODCAST w/o flow	0.084	1.081	5.536	0.196	0.920	0.963	0.980
FLODCAST	**0.074**	**0.843**	**4.965**	**0.169**	**0.936**	**0.971**	**0.984**
Mid term $k = 10$							
	Lower is better ↓				Higher is better ↑		
Method	AbsRel	SqRel	RMSE	RMSE-Log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Copy last	0.304	5.006	8.319	0.517	0.511	0.781	0.802
Qi et al. [14]	0.224	3.015	7.661	0.394	0.718	0.857	0.881
Hu et al. [11]	0.195	1.712	**6.375**	0.299	0.735	0.896	0.928
Sun et al. [4]	0.259	4.115	7.842	0.428	0.695	0.817	0.842
Goddard et al. [41]	0.211	2.478	7.266	0.357	0.724	0.853	0.882
DeFNet [12]	0.192	1.719	6.388	0.298	0.742	0.900	0.927
FLODCAST w/o flow	0.130	2.103	7.525	0.320	0.863	0.931	0.959
FLODCAST	**0.112**	**1.593**	6.638	**0.231**	**0.891**	**0.947**	**0.969**

both at short-term ($k = 5$, after 0.29 s) and at mid-term ($k = 10$, after 0.59 s).

Since there is no official evaluation protocol for depth forecasting on Cityscapes and considering the statistics in the training set (see Fig. 2), in which pixel occurrences strongly decrease as the depth increase, we clip values at 80 m as done in prior work for depth estimation [37,40].

For our experiments, we evaluate predictions using the same protocol of [37], i.e. by cropping out the bottom 20% of the image to remove the car hood, which is visible in every frame, then we rescale the frames at 256 × 512. In addition, we mask out ground truth pixels that are farther than the 80 m threshold.

We compare our approach with existing methods [11,12,14]. We also consider the depth estimation method of [41], which is adapted to depth forecasting through a multi-scale F2F [3] before the decoder, and the future instance segmentation model [4] adapted to generate future depth estimation of the predicted features, as previously done in [12]. We also report the trivial *Copy last* baseline [12], as a lower bound. Quantitative results for depth forecasting are reported in Table 1.

We exceed all the previous methods at short-term and mid-term predictions. Specifically, we beat all the existing approaches at short-term by a large margin for all the metrics, also reporting the highest inlier percentage. At mid-term term we exceed all the state-of-the-art approaches, in terms of AbsRel and SqRel, including the recent DeFNet (−42% and −8%), which employs both RGB frames and optical flows, even considering the camera pose during the training. Differently from DeFNet, we exploit depth maps and optical flows as sources of information, since they provide complementary features related to motion and geometric structure of the scene by means of a recurrent network. We believe that FLODCAST is capable of detecting such clues by extrapolating features from past sequences, which also implicitly contains the camera motion, without training a pose estimation network conditioned to specific future frames, like in [12], that clearly limits the application to forecast depths only at corresponding future time steps. We report a slight drop in terms of RMSE at mid-term compared to [11] and [12], however we still achieve concrete improvements in terms of RMSE-Log, by reducing the error of 22%. This indicates that the relative depth consistency is much better preserved by our approach than by the competitors.

Using its recurrent nature, FLODCAST is capable to generate a sequence of depth maps in the future without temporal sub-sampling, i.e. by producing all the intermediate forecasting steps (not only the last one, as done in [12]). In dynamic scenarios, like an urban setting, this is particularly useful, since objects can appear and be occluded several times from one frame to another. Such behavior might not emerge from subsampled predictions.

Some qualitative results are shown in Figs. 3 and 4, respectively for short-term and mid-term predictions. FLODCAST learns to locate the region containing the vanishing point by assigning higher depth values. Moreover, we observed that missing depth map values coming from zeroed values in the ground truth frames are mostly predicted correctly. This underlines that FLODCAST is able to anticipate depth maps up to mid-range predictions while being highly accurate, even though some parts of the scene may not have been labeled, due to bad measurements or missing data.

4.5. Future flow estimation

We evaluate optical flow forecasting capabilities on Cityscapes, by following the protocol of [10]. Therefore, we calculate the average end-point error EPE, according to Eq. (11), for the $t + 10$ frame (i.e. 0.59 s ahead), namely corresponding at the 20th frame for each val sequence. We carry out experiments at the resolution 256 × 512, by doubling the resolution, and we compare our approach with existing works, FAN [10] and OFNet [9], and some baselines from [10], namely (i) warping the flow field using the optical flow in each time step (namely *Warp Last*) and (ii) simply copying the one last (namely *Copy Last*).

Since our work is capable to provide optical flows for multiple future scenarios, we also assess our performance for every intermediate frames up to $t + 10$, by following the evaluation protocol in [9]. Thus, we measure the quality of our predictions generated autoregressively for each time step, by computing the mean squared error for u and v components and averaging them, according to Eq. (10). We report our quantitative results in Table 2.

We mainly found that the FLODCAST error drastically decreases over time. This brings us some considerations. First of all, FLODCAST combines different modalities, also exploiting spatio-temporal information, and that comes to be crucial to reduce the accumulation error through time. Because optical flow and depth maps are complementary each other, the model can better identify specific patterns, e.g. discriminating object motions at different resolutions in advance (see Fig. 8). This also allows to directly generate multiple future optical flows at a time with a shorter input sequence (i.e. $T = 3$ for FLODCAST while

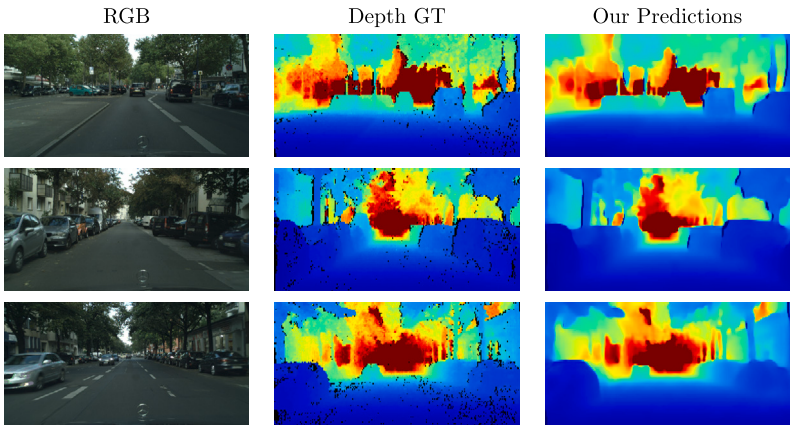

Fig. 3. Visualization results of future predictions on Cityscapes test set at short-term (k = 5). Black pixels in the ground truth (second column) are invalid measurements.

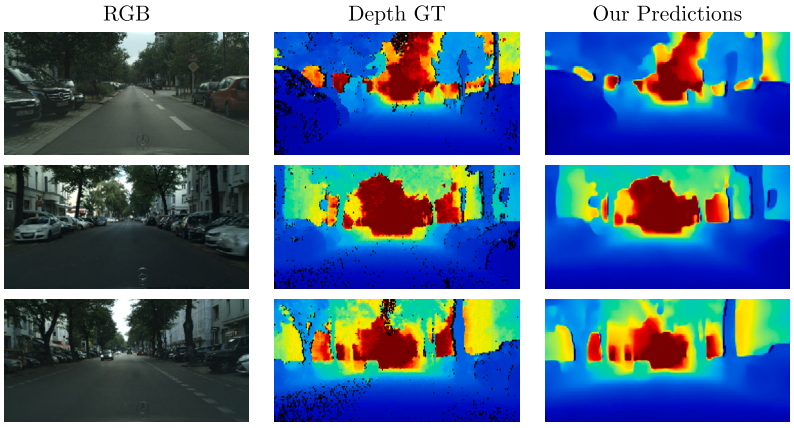

Fig. 4. Visualization results of future predictions on Cityscapes test set at mid-term (k = 10). Black pixels in the ground truth (second column) are invalid measurements.

Table 2
Quantitative results for flow forecasting on Cityscapes val set. In bold the lowest error. We denote with the symbol "–" if the corresponding result is not available or reproducible.

Method	MSE ↓										EPE ↓
	t + 1	t + 2	t + 3	t + 4	t + 5	t + 6	t + 7	t + 8	t + 9	t + 10	t + 10
Copy Last [10]	–	–	–	–	–	–	–	–	–	–	9.40
Warp Last [10]	–	–	–	–	–	–	–	–	–	–	9.40
FAN [10]	–	–	–	–	–	–	–	–	–	–	6.31
OFNet [9]	**0.96**	0.94	1.30	1.40	1.78	1.88	2.16	2.38	2.88	2.66	2.08
FLODCAST w/o depth	0.98	**0.80**	1.11	1.20	1.38	1.48	1.72	1.78	2.18	1.92	1.48
FLODCAST (Ours)	1.06	0.84	**1.10**	**1.12**	**1.34**	**1.44**	**1.62**	**1.68**	**2.12**	**1.74**	**1.38**

$T = 6$ for OFNet). Moreover, we found a substantial diminishing of the MSE up to 33% at $t + 10$ and that also supports our observations. Considering that OFNet has more supervision during training, i.e. it forecasts an output sequence shifted by one step ahead with respect to its input, this is the reason we believe performances are sometimes better at the beginning steps but then the error increases compared to FLODCAST.

In absence of intermediate results of MSE for other methods (i.e. FAN, for which no source code and models are available, as denoted in Table 2), we compare the overall performance by evaluating the EPE error at $t + 10$, also against the Flow Anticipating Network (FAN) proposed in [10], that generates future flows in a recursive way, by using the finetuned version of their model, which is learned to predict the flow for the single future time step given the preceding frames and the corresponding segmentation images.

We found remarkable improvements even at $t + 10$, by reducing the EPE with respect to FAN and OFNet as well. This highlights our choice that using optical flow with depth maps is better for determining future estimates than with the semantic segmentations employed in FAN. Restricting to observing past optical flows to generate a future one, as done in OFNet, does not allow forecasting models to make reliable long-range predictions autoregressively. Further improvements are obtained when multiple frames are predicted at a time, as FLODCAST does. Then, we demonstrate that FLODCAST is more accurate in predicting unobserved motions far into the future, without requiring semantic data, that is typically harder to get labeled with respect to depth maps, which are directly obtained by using commercial devices like LiDARs or stereo rigs. We also observe that excluding the depth map from FLODCAST, flow performance is reduced, since EPE increases by 6.8%. Despite the hard task of anticipating flow motion without seeing future frames, FLODCAST exceeds all the previous works, and it is more robust when depth is stacked into the input data.

4.6. Ablation study

In order to understand how significant the flow and depth as data sources are for anticipating the future, we exclude one of the two inputs

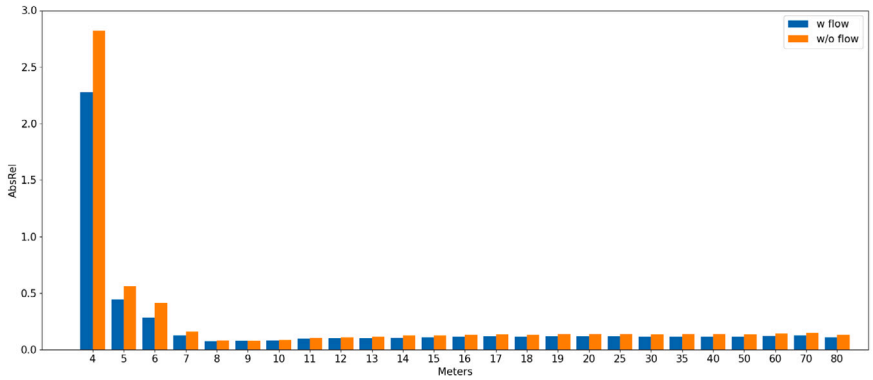

Fig. 5. Ablation study on depth forecasting in Cityscapes test set. We report the AbsRel error at $t+10$ per distance (in meters), both when the input data is composed of optical flows and depth maps (blue) or only depth (orange). Note that depth values below 3 m are not present in the test set. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

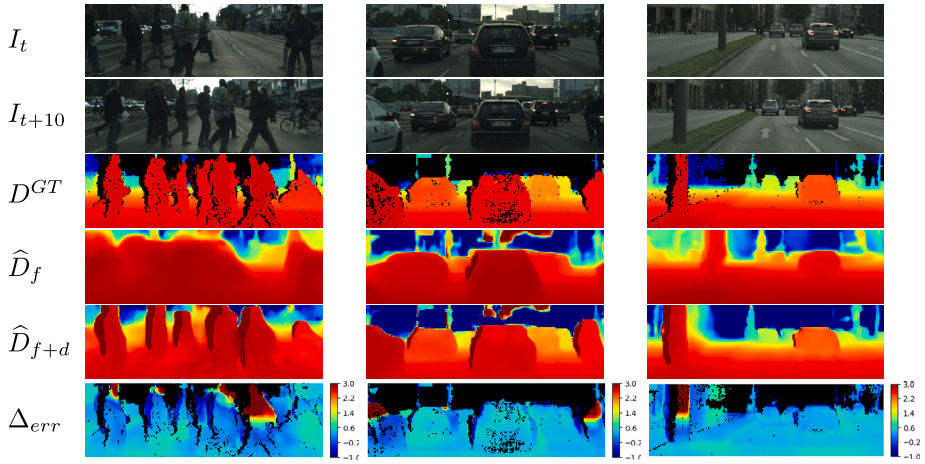

Fig. 6. Qualitative results of predicted depth maps of FLODCAST. The first two rows are the last observed frame I_t and the future one, I_{t+10}. The third row contains ground truth depth maps (D^{GT}) for the three samples. The 4th and 5th row show depth predicted without ($\hat{D}_f$) and with ($\hat{D}_{f+d}$) optical flows respectively. Pixel-wise difference in AbsRel errors between FLODCAST w/o flow and our FLODCAST (Δ_{err}) are depicted as heatmap plot in the 6th row. We report results for three different sequences in the Cityscapes test set.

at a time and we evaluate the performance compared with FLODCAST, which instead leverages both data sources.

Depth analysis. To demonstrate the importance of incorporating flow features for depth forecasting, we exclude optical flow from the input and we train FLODCAST using the $\mathcal{L}_{\text{depth}}$ loss (see Eq. (4)) to estimate future depth maps.

From Table 1 we observe that generating future depth maps through the past ones without leveraging optical flow as source data, i.e. FLODCAST w/o flow, worsens the predictions under all of the metrics. This points out the relevance of combining features extracted from past scenes, in terms of 2D motion and depth. Nonetheless, predicting only future depth maps using our approach, even discarding the optical flow information, gets improvements compared to prior works such as [11, 12]. At short-term $t+5$ FLODCAST w/o flow is the second best result overall, by reducing the errors by a large margin (e.g. AbsRel and SqRel respectively −53% and −27% from Hu et al. and −52% and −16% from DeFNet) with also higher percentage of inliers. At mid-term $t+10$ we reported drops of performance of FLODCAST w/o flow still limiting the AbsRel error and getting higher accuracy of inlier pixels. Overall, removing optical flow from the input data, FLODCAST still works better than all the existing works on forecasting unseen scenarios but then the lack of the information affects the performance for farther frames. In addition, we compute the AbsRel error distribution of FLODCAST, when depth maps are predicted through only optical flows (orange bars) or employing our multimodal approach (blue bars) and we plot a histogram at $t+10$ as function of the distance (Fig. 5).

We found notable improvements within 10 m when optical flow is part of the input. This is crucial in terms of safety since objects moving around a self-driving agent can be better defined according to their predicted distances. Indeed, from an ego-vehicle perspective, parts of the scene close to the observer are more likely to change over time. Considering that we are forecasting the depth for the whole image, just a few regions move considerably, corresponding to dynamic objects. The rest of the scene, typically the background, like buildings or vegetation, exhibits instead a static behavior and does not change much depth-wise even in presence of ego-motion. Therefore, the depth estimated for those far away pixels contains little error and, consequently, the tails of the two plots tend to be quite similar. Considering that the histogram represents depth errors 10 frames after from the last observed one, our FLODCAST is robust also for long distance when optical flow is part of the input. This also motivates our design choices of sticking data in a multimodal and multitasking approach.

We further provide some qualitative results in Fig. 6, so to underline how the contribution coming from the flow features is significant in generating very accurate depth maps, especially on moving objects, like pedestrians and vehicles. It is noteworthy that 2D motion displacements in the scene help to correctly predict depth values on different moving objects close to each other, e.g. pedestrians crossing the street, whose estimated depths collapse in a unique blob when optical flow is not taken into account. The same happens for cars at different distances from the camera, where their predicted depths look lumped together. That suggests that the model without flow features is less capable of distinguishing single instances.

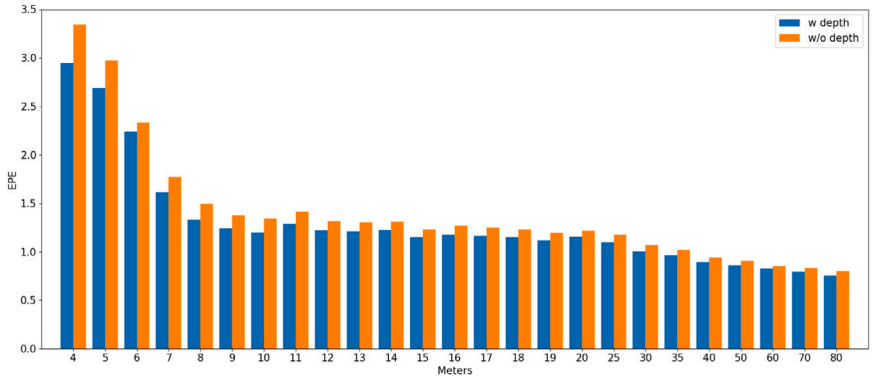

Fig. 7. Ablation study on flow forecasting in the Cityscapes test set. We report the EPE error at $t+10$ according to the distance (meters) of optical flows predicted by FLODCAST, in case of the input data being both optical flows and depth maps (blue) or only optical flows (orange). Note that depth values below 3 m are not present in the test set. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

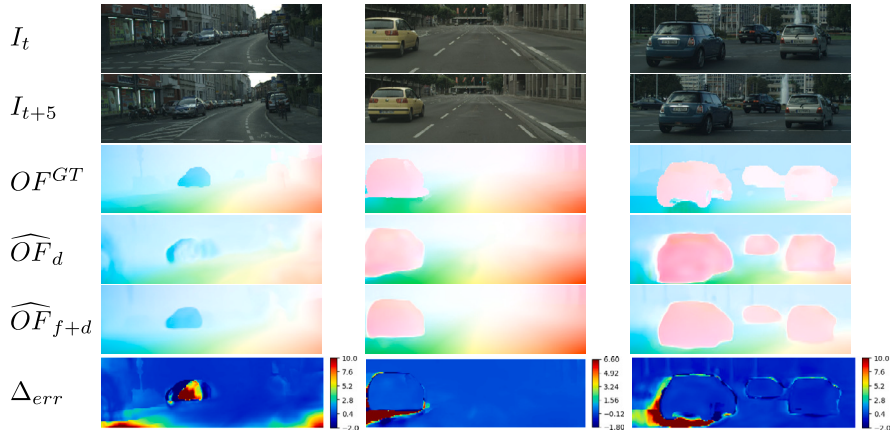

Fig. 8. Qualitative results of predicted optical flow maps of FLODCAST. The first two rows are the last observed frame I_t and the future one, I_{t+5}. The third row contains ground truth optical flow (OF^{GT}) for the three samples. The 4th and 5th rows show optical flow predicted without ($\widehat{OF}_d$) and with ($\widehat{OF}_{f+d}$) depth respectively. Pixel-wise difference in optical flow MSE between FLODCAST w/o depth and our FLODCAST (Δ_{err}) are depicted as heatmap plot in the 6th row. We report results for three different sequences in the Cityscapes test set.

Flow analysis. We discard depth maps from the input data and we train the network to predict future optical flows, i.e. by exploiting past flow features, while keeping the same $\mathcal{L}_{\text{flow}}$ loss (see Eq. (3)). We measure the optical flow predictions generated autoregressively for each time step, by computing the mean squared error on both the two flow channels and averaging them (Eq. (10)). From the flow forecasting results reported in Table 2, we observe that features extracted from both the optical flows and depth maps contribute to reduce the MSE errors on predicted flows, resulting in overall improvements after the first steps up to at $t+10$, i.e. +33% over OFNet and +9% over FLODCAST w/o depth, which is significant considering the high uncertainty for farther future scenarios. Compared with OFNet, FLODCAST w/o depth has the FlowHead module (as depicted in Fig. 1), in which specialized weights of convolutional layers are end-to-end trained in order to directly generate multiple optical flows at a time. Despite the notable reduction of the error through time, FLODCAST overcomes its performance when depth maps are included in the source data, which points out the importance of our multimodal approach. Looking at the last prediction, i.e. at $t+10$, FLODCAST w/o depth still exceeds other approaches, but reports an increase of the EPE error by +7% with respect to our multimodal approach. This fact suggests that recurrent architectures can achieve good results for forecasting tasks and they can improve if they are multimodal. In addition, we study the EPE error distribution according to distance. To do that, we collect all the predicted flows upsampled to 256×512 at $t+10$ on the test set, and we compute the error (see Eq. (11)) for all the pixels falling into the corresponding distance-based bins and we represent their averages in Fig. 7. Here, orange bars are errors reported by only using optical flow in input, while the blue ones incorporate also depth maps, i.e. our proposed FLODCAST model.

As can be seen in Fig. 7, the overall trend of EPE is to decrease as the depth increases. This is due to the fact that, parts of the scene far enough from the camera typically produce similar small motion, like objects moving at the background or static parts that are mainly affected by the camera motion, thus the predicted optical flows for such pixels are likely to be more accurate. Instead, pixels closer to the camera tend to have a more pronounced motion and that affects the predictions, especially of farther frames. We observe that EPE errors of FLODCAST are always lower when depth maps are provided as input (blue bars) than only using optical flow as unique data source (orange bars). In particular, we gain more within 15 m, which is the most relevant part of the scene concerning the safety and the drive planning of autonomous agents in very dynamic scenarios like the urban one. FLODCAST with depth maps has the potential to better disambiguate motions of pixels close to the observer than the far ones and vice versa.

Hence, flow forecasting results are more precise as long as the depth map is included in the input data. Based on this consideration, we reported in Fig. 8 some qualitative results on the Cityscapes test set, where we illustrate the ground truth optical flow in comparison with the optical flows obtained from FLODCAST, both exploiting or not the

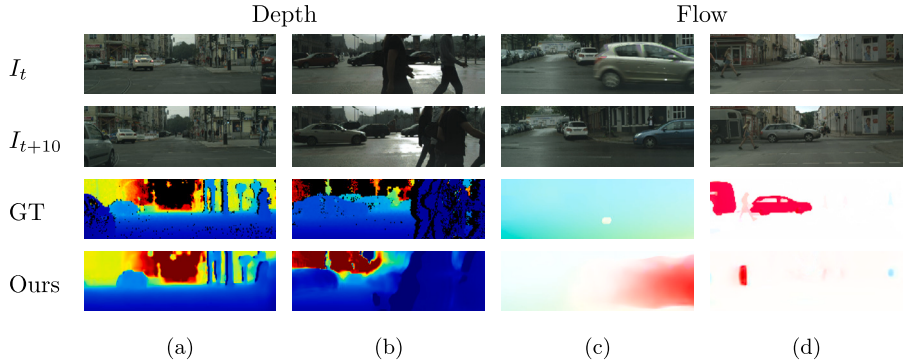

Fig. 9. Qualitative results of different failure cases at mid-term $t + 10$ for flow and depth forecasting. In particular, for each sequence we report the last observed frame (I_t) and the future one (I_{t+10}), as well as the groundtruth (GT) and ours predictions.

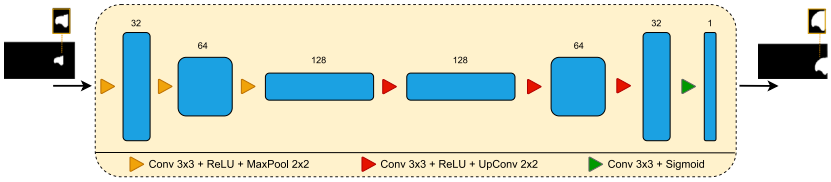

Fig. 10. Denoising autoencoder (DAE) used to refine the generated future instance segmentation masks. The model is based on a convolutional encoder–decoder structure, where the encoder compresses the input into the latent space and the decoder gradually upsamples the features back to the original image size.

depth map as an additional input source. Finally, we show the heatmaps in the last row of Fig. 8 of the MSE errors with respect to the ground truth as differences between the predictions generated by FLODCAST without depth map and by FLODCAST using both data sources. Specifically, we report enhancements mostly on moving objects, whose shapes are more correctly defined, as shown in the red parts of the cars and the light blue around their shapes.

4.7. Performance details

To take into account the forecasting problem in terms of anticipation, predictions have to be provided early. We therefore analyze the performance of FLODCAST at inference time. We test our model using a single NVIDIA RTX 2080. At runtime, FLODCAST requires 8.8 GB of GPU memory and it is able to forecast sequences of $K = 3$ consecutive depth maps and optical flows in 40 ms (25FPS). Our predictions are estimated for multiple frames ahead simultaneously, which is more efficient than making predictions for a single one, as done in [10,11,14].

4.8. Failure cases

To conclude our analysis, we present some failure cases. We report in Fig. 9 interesting failures for flow and depth forecasting at $t + 10$. We found that observed objects leaving the field of view during the predicted future timespan can cause issues. More importantly, the model is unable to forecast the presence of objects that were not originally observed in the past, as there is no relationship between past and unseen frames. For such cases, FLODCAST tends to replicate reasonable depth values according to spatio-temporal features in the past frames (e.g. see the car suddenly appearing on the left side in Fig. 9(a)). Likewise, the optical flow of moving objects tends to follow the expected future motion based on the observed scene, e.g. the car in Fig. 9(c).

We also found mistakes in case of severe occlusion, such as in the presence of crowds (Fig. 9(b)). Here the depth prediction appears to be smoothed out, losing the sharpness of the borders. In a few cases, FLODCAST also loses precision when modeling objects that entered just in the last frames of the sequence (e.g. the pedestrian walking in Fig. 9(d)).

5. Segmentation forecasting

We now show how FLODCAST can be employed to address downstream tasks such as forecasting segmentation masks. In fact, flow-based forecasting methods have demonstrated that warping past features onto future frames allows producing competitive semantic segmentations [8,9,13]. Since FLODCAST predicts dense optical flows in the future, we use the recent lightweight framework introduced in [9], to explore possible improvements on the segmentation forecasting problem as a downstream task through our predictions, in terms of binary instances and semantic categories. To this end, from the whole framework, which also includes a flow forecasting module, named OFNet, we only take MaskNet, which is a neural network that warps binary instances from the current frame onto the future one. Because MaskNet requires future optical flows to warp instances, we replace OFNet with FLODCAST, by only retaining our flow predictions and discarding depth maps.

In order to generate future predictions, both instance and semantic segmentations, we follow the same protocol training in [9]. We first finetune a MaskNet model pretrained on ground truth masks (the MaskNet-Oracle model from [9]), by feeding future optical flows predicted by FLODCAST. We perform separate trainings to make predictions up to $T+3$ (short-term) and $T+9$ frames ahead (mid-term).[1] We denote these two models as MaskNet-FC. Second, we study how binary instances predicted by MaskNet can be improved. Because we employ predicted optical flow to estimate future binary masks, motion mistakes may affect some pixels of the object to be warped. We also believe that

[1] Note that in the literature there is a slight misalignment when referring to short-term and mid-term, depending on the task. For depth and flow forecasting we refer to short-term mid-term as $T + 5$ and $T + 10$ and for segmentation forecasting as $T + 3$ and $T + 9$ respectively.

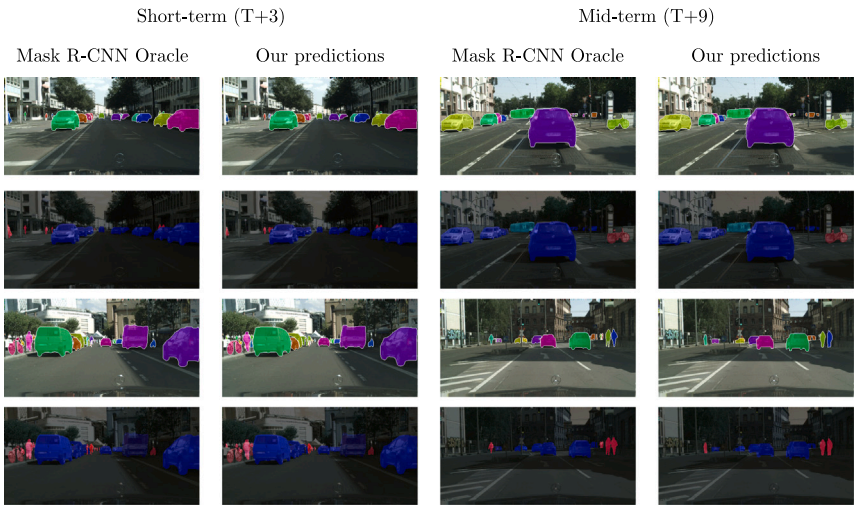

Fig. 11. Qualitative results of future instance and semantic segmentation predictions on the Cityscapes val set both at short-term and mid-term generated by MaskNet-FC+DAE.

Table 3
Future instance segmentation (AP and AP50) and future semantic segmentation (IoU) of moving objects on the Cityscapes val set. Best results in bold, second best underlined.

Method	Short term (T + 3)			Mid term (T + 9)		
	AP	AP50	IoU	AP	AP50	IoU
Mask RCNN oracle	34.6	57.4	73.8	34.6	57.4	73.8
MaskNet-Oracle [9]	24.8	47.2	69.6	16.5	35.2	61.4
Copy-last segm. [3]	10.1	24.1	45.7	1.8	6.6	29.1
Optical-flow shift [3]	16.0	37.0	56.7	2.9	9.7	36.7
Optical-flow warp [3]	16.5	36.8	58.8	4.1	11.1	41.4
Mask H2F [3]	11.8	25.5	46.2	5.1	14.2	30.5
F2F [3]	19.4	39.9	61.2	**7.7**	19.4	41.2
MaskNet [9]	**19.5**	**40.5**	**65.9**	6.4	18.4	45.5
MaskNet-FC	18.1	37.8	65.4	6.7	18.9	48.4
MaskNet-FC+DAE (Ours)	18.3	39.0	65.7	7.1	**20.7**	**49.2**

some drops in the performance of MaskNet are due to misleading pixels, that are badly labeled as background instead of instance and vice versa. This effect is more pronounced when an object appears smaller and its predicted flow is not accurate. Inspired by [42], we address this issue by introducing a Denoising AutoEncoder network (shortened to DAE) to the output of MaskNet, so to make binary masks cleaner and to make them as much aligned as possible to the ground truth. The network, depicted in Fig. 10, has an encoder consisting of Conv-ReLU-MaxPool sequences with 32, 64 and 128 filters, and a decoder where Conv-ReLU-UpSample operations are used with 128, 64 and 32 filters. The output is generated after a convolution operation with a single channel, 3×3 kernel filter and a sigmoid activation function. At inference, outputs are binarized using a 0.5 threshold.

Because MaskNet warps object instances based on optical flows, the generated masks have to be fed to the DAE to get refined. Therefore, we train the DAE, by using autoregressive flows and freezing MaskNet pretrained weights. Specifically, we train DAE for 3 epochs with a per-pixel MSE loss function with predicted flows up to 3 frames ahead (i.e. $T+3$, short-term). We observe that using a Dice loss [43] (already employed to train MaskNet), even in combination with the L2 loss, DAE performs worse than with the MSE function. We believe that is due to the fact that further improvements on instance shapes are not always possible with region-based losses (like Dice loss), instead MSE is more suitable to binarize an instance as a whole image. We continue to finetune the DAE for 3 more epochs using the autoregressive flows predicted up to 9 frames ahead (i.e. $T+9$, mid-term) to adapt the network to less accurate inputs. Doing so, we are able to provide a single autoencoder trained to refine instances, which are generated by MaskNet through autoregressive flows predicted up to 9 frames ahead. Hence, our overall segmentation forecasting architecture, i.e. MaskNet-FC+DAE, is obtained by appending the DAE to the MaskNet mid-term model. This architecture allows to utilize a unique segmentation model to generate future instance segmentation up to 9 frames ahead.

We conduct experiments on the Cityscapes val set, generating future instance and semantic segmentations of 8 different categories of moving objects, both 3 frames and 9 frames ahead (up to about 0.5 s later) as done in [3], respectively referred to in the literature as short-term and mid-term. We use the mAP and mAP50 metrics for instance segmentation, and mIoU (mean IoU) for semantic segmentation. We show our quantitative results in Table 3.

We report segmentation results achieved by MaskNet [9], using flows predicted by our FLODCAST, also considering the denoising autoencoder (DAE), proposed to refine warped masks. We compare our results with the original flow-based approach MaskNet [9]. We also report the oracle reference, where a Mask RCNN [44] is used directly on future frames, as well as MaskNet-Oracle whose model is our upper bound flow-based approach since segmentations are warped using ground truth flows. Moreover, we listed the performances of 4 simple baselines and the commonly used F2F approach [3].

We found that MaskNet, using flows predicted by FLODCAST, improves at mid-term, getting +0.5% and +2.9%, respectively for instance and semantic segmentations compared to the original formulation of [9]. Meanwhile, we observe a negligible drop at short-term, since FLODCAST generates more accurate flows after the first iteration. Because the segmentation performance typically degrades over the time, we pay attention to the impact of appending our DAE at the end of MaskNet to enhance instance and semantic results mainly at mid-term (i.e. 9 frames ahead, 0.5 s), which is a more challenging scenario

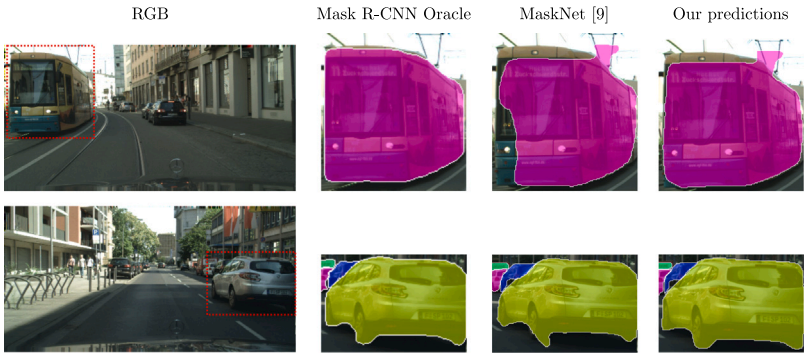

Fig. 12. Some qualitative results at short-term (T+3) of future instance segmentation predictions on the Cityscapes val set.

Fig. 13. Some qualitative results at mid-term (T+9) of future instance segmentation predictions on the Cityscapes val set.

than the short-term one. When the DAE is trained to refine instance masks up to mid-term we report a considerable improvement against the F2F approach with a gain of +1.3% in AP50 and +8% in IoU. Some qualitative results of future instance and semantic segmentation are shown in Fig. 11.

We additionally provide some qualitative results in terms of instance segmentations predicted, by using FLODCAST and MaskNet-FC+DAE, in comparison with the previous framework, i.e. OFNet and MaskNet. We show enhancements on different objects and shapes predicted both at short-term (Fig. 12) and mid-term (Fig. 13), such as the big shapes (like trams and trucks) as well as some details (like car wheels and pedestrians on the ground).

6. Conclusions

In this work, we proposed FLODCAST, a novel multimodal and multitask network able to jointly forecast future optical flows and depth maps using a recurrent architecture. Differently from prior work, we forecast both modalities for multiple future frames at a time, allowing decision-making systems to reason at any time instant and yielding state-of-the-art results up to 10 frames ahead on the challenging Cityscapes dataset. We demonstrated the superiority of exploiting both optical flow and depth as input data against single-modality models, showing that leveraging both modalities in input can improve the forecasting capabilities for both flow and depth maps, especially at farther time horizons. We also demonstrated that FLODCAST can be applied on the downstream task of segmentation forecasting, relying on a mask-warping architecture, improved with a refining instance model that boosts mid-range predictions.

Further research will be considered for future developments, which include the usage of a transformer architecture to boost our multitasking model. Other lines of research may also include more performing mask-level segmentation models to be trained end-to-end with a flow forecasting architecture, in order to directly perform the task for multiple frames at a time, in the same sense FLODCAST was designed.

CRediT authorship contribution statement

Andrea Ciamarra: Software, Writing – original draft. **Federico Becattini:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision. **Lorenzo Seidenari:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing, Methodology. **Alberto Del Bimbo:** Funding acquisition, Project administration, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data used in the paper is publicly available.

Acknowledgments

This work was supported by the European Commission under European Horizon 2020 Programme, grant number 951911—AI4Media. This work was partially supported by the Piano per lo Sviluppo della Ricerca (PSR 2023) of the University of Siena, Italy - project FEATHER: Forecasting and Estimation of Actions and Trajectories for Human–robot intERactions.

References

[1] T. Salzmann, B. Ivanovic, P. Chakravarty, M. Pavone, Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data, in: European Conference on Computer Vision, Springer, 2020, pp. 683–700.
[2] F. Marchetti, F. Becattini, L. Seidenari, A. Del Bimbo, Smemo: social memory for trajectory forecasting, 2022, arXiv preprint arXiv:2203.12446.
[3] P. Luc, C. Couprie, Y. Lecun, J. Verbeek, Predicting future instance segmentation by forecasting convolutional features, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 584–599.
[4] J. Sun, J. Xie, J.-F. Hu, et al., Predicting future instance segmentation with contextual pyramid convlstms, in: Proceedings of the 27th Acm International Conference on Multimedia, 2019, pp. 2043–2051.
[5] J.-F. Hu, J. Sun, Z. Lin, et al., APANet: Auto-path aggregation for future instance segmentation prediction, IEEE Trans. Pattern Anal. Mach. Intell. (2021).
[6] Z. Lin, J. Sun, J.-F. Hu, Q. Yu, et al., Predictive feature learning for future segmentation prediction, in: Proceedings of International Conference on Computer Vision, 2021, pp. 7365–7374.
[7] P. Luc, N. Neverova, C. Couprie, et al., Predicting deeper into the future of semantic segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 648–657.
[8] J. Saric, M. Orsic, T. Antunovic, et al., Warp to the future: Joint forecasting of features and feature motion, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2020, pp. 10648–10657.
[9] A. Ciamarra, F. Becattini, L. Seidenari, A. Del Bimbo, Forecasting future instance segmentation with learned optical flow and warping, in: International Conference on Image Analysis and Processing, Springer, 2022, pp. 349–361.
[10] X. Jin, H. Xiao, X. Shen, et al., Predicting scene parsing and motion dynamics in the future, Adv. Neural Inf. Process. Syst. 30 (2017).
[11] A. Hu, F. Cotter, N. Mohan, et al., Probabilistic future prediction for video scene understanding, in: European Conference on Computer Vision, Springer, 2020, pp. 767–785.
[12] S. Nag, N. Shah, A. Qi, R. Ramachandra, How far can I go?: A self-supervised approach for deterministic video depth forecasting, 2022, arXiv preprint arXiv:2207.00506.
[13] A. Terwilliger, G. Brazil, X. Liu, Recurrent flow-guided semantic forecasting, in: 2019 IEEE Winter Conference on Applications of Computer Vision, WACV, IEEE, 2019, pp. 1703–1712.
[14] X. Qi, Z. Liu, Q. Chen, J. Jia, 3D motion decomposition for RGBD future dynamic scene synthesis, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2019, pp. 7673–7682.
[15] A. Mertan, D.J. Duff, G. Unal, Single image depth estimation: An overview, Digit. Signal Process. (2022) 103441.
[16] D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2650–2658.
[17] J. Xie, C. Lei, Z. Li, et al., Video depth estimation by fusing flow-to-depth proposals, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2020, pp. 10100–10107.
[18] M. Ranzato, A. Szlam, J. Bruna, et al., Video (language) modeling: a baseline for generative models of natural videos, 2014, arXiv preprint arXiv:1412.6604.
[19] M. Mathieu, C. Couprie, Y. LeCun, Deep multi-scale video prediction beyond mean square error, 2015, arXiv preprint arXiv:1511.05440.
[20] N. Kalchbrenner, A. Oord, K. Simonyan, et al., Video pixel networks, in: International Conference on Machine Learning, PMLR, 2017.
[21] J. Van Amersfoort, A. Kannan, M. Ranzato, et al., Transformation-based models of video sequences, 2017, arXiv preprint arXiv:1701.08435.
[22] Y.-H. Kwon, M.-G. Park, Predicting future frames using retrospective cycle gan, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2019, pp. 1811–1820.
[23] H. Boulahbal, A. Voicila, A. Comport, Forecasting of depth and ego-motion with transformers and self-supervision, 2022, arXiv preprint arXiv:2206.07435.
[24] Z. Tu, W. Xie, D. Zhang, et al., A survey of variational and CNN-based optical flow techniques, Signal Process., Image Commun. 72 (2019).
[25] M. Zhai, X. Xiang, N. Lv, X. Kong, Optical flow and scene flow estimation: A survey, Pattern Recognit. 114 (2021) 107861.
[26] E. Ilg, N. Mayer, T. Saikia, et al., Flownet 2.0: Evolution of optical flow estimation with deep networks, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2017, pp. 2462–2470.
[27] D. Sun, X. Yang, M.-Y. Liu, J. Kautz, Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume, in: Proceedings of Computer Vision and Pattern Recognition, 2018, pp. 8934–8943.
[28] Z. Teed, J. Deng, Raft: Recurrent all-pairs field transforms for optical flow, in: European Conference on Computer Vision, Springer, 2020, pp. 402–419.
[29] Z. Huang, X. Shi, C. Zhang, et al., FlowFormer: A transformer architecture for optical flow, 2022, arXiv preprint arXiv:2203.16194.
[30] X. Shi, Z. Huang, D. Li, et al., Flowformer++: Masked cost volume autoencoding for pretraining optical flow estimation, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2023, pp. 1599–1610.
[31] Y. Lu, Q. Wang, S. Ma, et al., Transflow: Transformer as flow learner, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2023, pp. 18063–18073.
[32] Z. Yin, J. Shi, Geonet: Unsupervised learning of dense depth, optical flow and camera pose, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1983–1992.
[33] S. Vedula, S. Baker, P. Rander, R. Collins, T. Kanade, Three-dimensional scene flow, in: Proceedings of the Seventh IEEE International Conference on Computer Vision, Vol. 2, IEEE, 1999, pp. 722–729.
[34] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015.
[35] I. Laina, C. Rupprecht, V. Belagiannis, et al., Deeper depth prediction with fully convolutional residual networks, in: 2016 Fourth International Conference on 3D Vision, 3DV, IEEE, 2016, pp. 239–248.
[36] M. Cordts, M. Omran, S. Ramos, et al., The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.
[37] A. Pilzer, D. Xu, M. Puscas, et al., Unsupervised adversarial depth estimation using cycled generative networks, in: 2018 International Conference on 3D Vision, 3DV, IEEE, 2018, pp. 587–595.
[38] D. Eigen, C. Puhrsch, R. Fergus, Depth map prediction from a single image using a multi-scale deep network, Adv. Neural Inf. Process. Syst. 27 (2014).
[39] S. Baker, D. Scharstein, J. Lewis, et al., A database and evaluation methodology for optical flow, Int. J. Comput. Vis. 92 (2011).
[40] V. Casser, S. Pirk, R. Mahjourian, A. Angelova, Unsupervised monocular depth and ego-motion learning with structure and semantics, in: Proceedings of Conference on Computer Vision and Pattern Recognition Workshops, 2019.
[41] C. Godard, O. Mac Aodha, M. Firman, G.J. Brostow, Digging into self-supervised monocular depth estimation, in: Proceedings of International Conference on Computer Vision, 2019, pp. 3828–3838.
[42] A.J. Larrazabal, C. Martinez, E. Ferrante, Anatomical priors for image segmentation via post-processing with denoising autoencoders, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2019, pp. 585–593.
[43] F. Milletari, N. Navab, S.-A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: 2016 Fourth International Conference on 3D Vision, 3DV, IEEE, 2016, pp. 565–571.
[44] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017.

Andrea Ciamarra achieved a master's degree in computer engineering cum laude in 2019 from the University of Florence, with a thesis related to vehicle motion predictions through optical flow. Currently, he is a Ph.D. student at Media Integration and Communication Center and working on forecasting tasks for behavior anticipation in automotive.

Federico Becattini is a Tenure-Track Assistant Professor at the University of Siena. His research focuses on computer vision and memory-based learning. He organized tutorials and workshops at ICPR2020, ICIAP2020, ACMMM2022, ICPR2022, ECCV2022. He has co-authored more than 40 papers. He is Associate Editor of the International Journal of Multimedia Information Retrieval (IJMIR).

Lorenzo Seidenari is an Associate Professor at the University of Florence. His research focuses on deep learning for object and action recognition. He is an ELLIS scholar. He was a visiting scholar at the University of Michigan in 2013. He gave a tutorial at ICPR 2012 on image categorization. He is author of 16 journal papers and more than 40 peerreviewed conference papers.

Alberto Del Bimbo is an Emeritus Professor and was director of the Media Integration and Communication Center at University of Florence. His interests are multimedia and computer vision. He was General Co-Chair of ACMMM2010, ECCV2012, ICPR2020. ACM nominated him Distinguished Scientist and he received the SIGMM Technical Achievement Award for Outstanding Technical Contributions to Multimedia Computing, Communications and Applications.