FlowNetU: Accurate Uncertainty Estimation of Optical Flow for Video Object Detection

JUN-GU KANG, Dept. of Electronic Engineering, Hanyang University, South Korea
SI-DONG ROH, Dept. of Electronic Engineering, Hanyang University, South Korea
KI-SEOK CHUNG, Dept. of Electronic Engineering, Hanyang University, South Korea

Video object detection (VOD) is a challenging task to resolve ambiguities owing to various issues such as motion blur and occlusion. Although various types of ambiguities will take place per pixels in an image, flow fields make equal contributions for VOD across the image. This may increase false positive (FP) results. In this paper, we propose a method that utilizes motion uncertainty for VOD. The trained optical flow estimation model helps detector to suppress unreliable flow fields in order to avoid misaggregation which causes mislocalization. Our proposed method improves mean average precision by 1.27 and decreases the FP rate by 10.59%. This verifies that utilizing motion uncertainty for video recognition tasks is very effective.

CCS Concepts: • **Computing methodologies** → **Object detection**.

Additional Key Words and Phrases: Uncertainty, Video Object Detection, Optical Flow Estimation

ACM Reference Format:
Jun-Gu Kang, Si-Dong Roh, and Ki-Seok Chung. 2021. FlowNetU: Accurate Uncertainty Estimation of Optical Flow for Video Object Detection. 1, 1 (September 2021), 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Today, deep learning has actively been utilized in various vision tasks. Particularly, deep-learning-based object detection has been adopted in many fields such as autonomous driving and CCTV surveillance where highly reliable operation is required. For instance, an autonomous vehicle should be able to detect cars, pedestrians, bicycles or traffic lights with a high accuracy. Typically, to improve accuracy, multiple types of sensors are employed. Among them, the LiDAR sensor can detect the object accurately by generating precise three-dimensional images. However, the cost of the sensor is relatively high and performance degrades considerably in an adverse weather condition. Therefore, today, the camera sensor is arguably the most commonly used sensor for object detection [24]. Camera sensors typically take temporal series of images as input. Using a single image detector (e.g., R-FCN [7]) for these video data may show an unacceptably-poor performance owning to video-specific ambiguity issues such as occlusion, motion blur, rare pose, and defocus, etc. To resolve such issues, in video object detection (VOD), temporal correlation among successive images may be taken into consideration to take advantages of the appearances in neighboring frames.

Optical flow, the pattern of pixel-wise motions between two consecutive image frames, is commonly regarded as an excellent method to model the temporal correlation. Several studies proposed the optical-flow-based VOD [22, 26, 27].

Authors' addresses: Jun-Gu Kang, gjk6626@hanyang.ac.kr, Dept. of Electronic Engineering, Hanyang University, Seoul, South Korea; Si-Dong Roh, sdroh1027@hanyang.ac.kr, Dept. of Electronic Engineering, Hanyang University, Seoul, South Korea; Ki-Seok Chung, kchung@hanyang.ac.kr, Dept. of Electronic Engineering, Hanyang University, Seoul, South Korea.

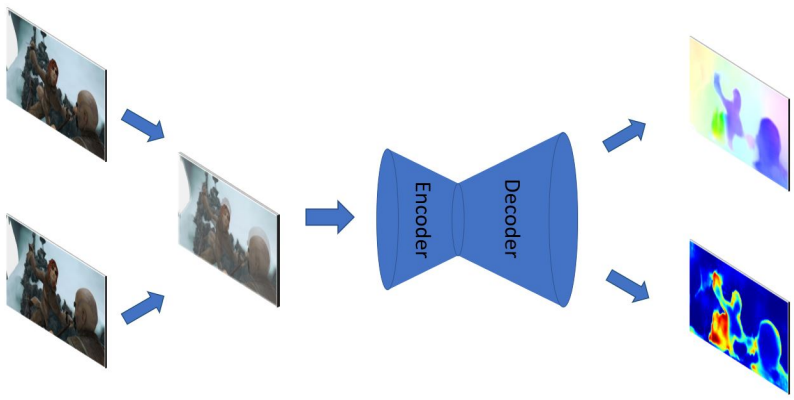

Fig. 1. The encoder-decoder network of FlowNetU. FlowNetU takes a concatenation of input images as input and estimates optical flow (above) and its variance (below). The variance is used for reliable VOD.

Specifically, they capture motion information among adjacent frames and detect objects that are difficult to recognize with only one image. Several conventional approaches were proposed to estimate an optical flow accurately [19, 25]. In a deep learning model for optical flow estimation, a network for predicting the optical flow should be trained [8, 13]. However, most existing methods overlook the importance of the level of uncertainty in the output. Unless the uncertainty is properly taken into consideration, the detector may fail to predict the motion of objects with erroneous flow fields, thereby typically leading to a high increase in the false positive (FP) rate.

To resolve this concern, we propose a method for optical flow estimation in this paper. In the proposed method called FlowNetU, the level of uncertainty is predicted, and this information is utilized in VOD. Our method assumes the distribution of flow fields to be Gaussian and estimates the uncertainty of each flow field vector. FlowNetU employs FlowNet [8] as the base model for optical flow estimation. FlowNetU is trained with a new loss function in contrast to the mean square error loss used in [8]. Specifically, a maximum likelihood estimation for Gaussian distribution to utilize the variance of flow fields is used as the new loss function. We show that our method predicts flow uncertainty more accurately than other existing methods on Sintel [5] and Middlebury [1] benchmarks. Additional experiment was carried out to confirm the effect of flow uncertainty. Both FlowNetS [8] and FlowNetU are used as the flow network of the VOD framework. The experimental results demonstrate that our approach offers a better detection accuracy and a lower false positive rate. For ImageNet VID, we improved mean average precision (mAP) by 1.27 and FP by -10.59%.

2 RELATED WORK

Optical Flow Estimation. Based on the mathematical formulation of the Horn-Schunck method [12], variational optimization is widely adopted to estimate optical flow [3, 19]. FlowNet [8] is the first optical flow estimation method using convolutional neural network (CNN). Quite a few models such as FlowNet2 [13] and PWC-Net [21] followed. These works measured only a deterministic flow field value. In contrast, we train an optical flow estimation model considering the distribution of the flow field and define the loss function to estimate uncertainty, which is based on FlowNet [8].

Uncertainty Estimation for Object Detection. Deep networks have achieved a great success in vision tasks. In spite of their high accuracy, there is a problem that the model tends to be overconfident about the result. This has raised the

need of confidence measure and calibration. The authors of [14] proposed a method to make vision task model robust to noisy data by measuring two types of uncertainty, aleatoric and epistemic. Aleatoric uncertainty is data-inherent uncertainty and epistemic uncertainty is uncertainty induced by the model itself. Aleatoric uncertainty is measured by calculating the mean and the variance from the output, while epistemic uncertainty is estimated by the distribution of the network's weight. A Bayesian neural network, e.g., Monte Carlo dropout [9], is used to compute the distribution.

There have been several works to measure the uncertainty in the object detection in a single image. The authors of [11] modeled the coordinates of bounding boxes as Gaussian distribution and calculated Kullback-Leibler (KL) divergence between the prediction and the ground-truth (GT) box. The authors of [6] also assumed that the location of bounding boxes should follow Gaussian distribution and achieved better performance than YOLOv3 [17]. Motivated by this work, we make an assumption that a flow field follows Gaussian distribution.

Object Detection in Video. Video is a series of successive images. While many single image detectors have achieved a high accuracy in the object detection task [7, 18], standalone video object detectors often suffer from errors due to various ambiguities as mentioned above. To overcome this weakness, optical flow is used to estimate the difference in object motions in consecutive frames. DFF [27] and FGFA [26] are motion-based VOD methods exploiting optical flow. In these methods, optical flow propagates the features of adjacent frames to reinforce the weak feature of the current frame. MANet [22] calculates the optical flow in the pixel-level calibration and helps detection in the instance-level calibration. In our experiment, we adopt FGFA as the base VOD framework. Contrary to existing works, our method takes into account uncertainty information inherent in flow fields to allow the detector to identify reliable flow fields.

3 METHOD

Our method proceeds in two steps. At first, FlowNetU is trained end-to-end with Gaussian modeling for optical flow. Then, VOD is performed with the trained FlowNetU. In this VOD stage, a new weight is introduced to improve the detection accuracy with the estimated uncertainty from FlowNetU.

Gaussian Modeling for Optical Flow. A CNN-based optical flow estimation model takes a pair of consecutive images $x = (I_1, I_2)$ as input and outputs the corresponding dense flow field $\hat{\mathbf{F}} = (u, v)$. Given dataset $D = \{(x, \mathbf{F}) | \mathbf{F} = (u^{GT}, v^{GT})\}$, the model learns a mapping function $g : x \rightarrow \mathbf{F}$. FlowNet [8] is the first work to learn the mapping function using an end-to-end trainable CNN architecture and achieves a good performance for optical flow estimation. In this work, we train FlowNetU which can estimate uncertainty of a flow field. FlowNetU is an encoder-decoder network which is based on [8]. The encoder network is a series of convolution layers which will generate high-dimensional down-sampled features from input images.

As shown in Fig 2, the decoder network generates flow field vectors and variances for each pixel. In the decoder, the features pass through the deconvolution layers (blue *deconv* arrow) and the flow prediction layers (gray $flow$ arrow) generate the intermediate features. The intermediate features are concatenated by the skip-connected features from the corresponding layers in the encoder (green *conv* arrow) and the previously generated flow fields and variances (red arrow). Each flow field is used to calculate a multiscale loss. The multiscale loss assigns a smaller weight to the lower resolution feature and vice versa. In [8], the network is trained by minimizing the average Euclidean distance between each flow field and the GT which is called average endpoint error (AEPE).

$$AEPE = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(u_i - u_i^{GT})^2 + (v_i - v_i^{GT})^2} \tag{1}$$

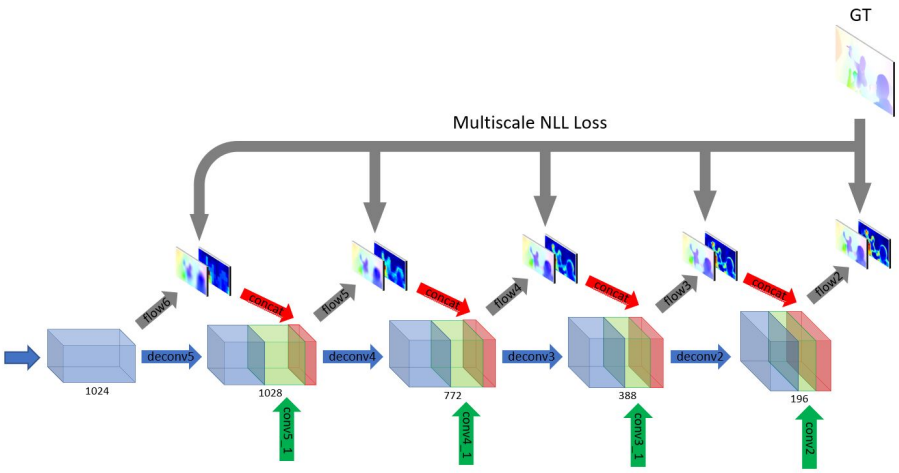

Fig. 2. The decoder architecture of FlowNetU. A deconvoluted feature (blue box), a skip-connected feature from the encoder (green box) and a predicted flow field and a variance (red box) are concatenated to become the next feature. The intermediate flow fields and variances are used to calculate the multiscale NLL loss with the ground-truth. The lower resolution flow fields are assigned with lower weights.

However, this loss function compares only the deterministic value of the flow field and the GT. Therefore the model does not know how reliable the output is. Instead of this AEPE loss, we design a new loss function to consider the uncertainty of the flow field. Specifically, we model the distribution of the flow field and use the variance of the distribution as the uncertainty as suggested in [14].

If $p(\mathbf{F}|x, D)$ is the true distribution of the flow field, the mean $p(\mathbf{F}|x, D)$ will be a flow field value and the variance will be uncertainty. Since the flow field has two mean values, μ_x and μ_y for x and y-direction, respectively, the variance also has two values, Σ_x^2 and Σ_y^2 for both directions. With these variables, the flow field can be modeled by Gaussian distribution $p(\mathbf{F}|x) = N(\mathbf{F}; \mu_x, \mu_y, \Sigma_x^2, \Sigma_y^2)$. To deliver the variance through the network, the deconvolution layers in the decoder are expanded by two additional channels. The objective of this task is to maximize the likelihood:

$$L(u^{GT}|\mu_x, \Sigma_x^2) = \frac{1}{\sqrt{2\Sigma_x^2}} \exp\left(-\frac{(u^{GT}-u)^2}{2\Sigma_x^2}\right) \tag{2}$$

Maximizing the equation (2) is identical to minimizing the negative log likelihood (NLL):

$$\begin{aligned} NLL &= -\log(L(u^{GT}|\mu_x, \Sigma_x^2) \\ &= \frac{(u^{GT}-u)^2}{2\Sigma_x^2} + \log \Sigma_x^2 \end{aligned} \tag{3}$$

A loss function for the y-direction flow can be defined similarly. The NLL loss consists of two terms: EPE term ($(u^{GT}-u)^2/2\Sigma_x^2$) and uncertainty term ($\log \Sigma_x^2$). When the variance goes high, the uncertainty term becomes significant while the EPE term is suppressed. This has effects that the model learns more from the pixels that may misguide the motion. Conversely, a low variance gives a bigger penalty to the model from the EPE term. As a result, the EPE term becomes more significant since the model is confident about the output.

Flow Field Uncertainty Weight. We claim that the accuracy of the video object detection is improved by applying a

weight with uncertainty predicted by FlowNetU. As mentioned above, we use FGFA [26] as the base VOD framework. We construct a modified VOD framework which is called FlowNetU-FGFA. The FlowNetU-FGFA framework is composed mainly of four trainable networks according to [26]. First, the feature network extracts features from a reference frame and the current frame. We denote the reference frame by I_j and the current frame I_i. Next, the flow network $\mathcal{F}$ computes the flow field $\mathcal{M}_{j\to i}$ between the two frames.

$$\mathcal{M}_{j\to i} = \mathcal{F}(I_i, I_j) \tag{4}$$

The feature of the reference frame is warped by a warping function $\mathcal{W}$,

$$f_{j\to i} = \mathcal{W}(f_j, \mathcal{M}_{j\to i}) \tag{5}$$

The warping function $\mathcal{W}$ is a bilinear interpolation by default. The warped features are gathered according to cosine similarity. The embedding network $\mathcal{E}$ projects the warped features into a new dimension for similarity measure. A cosine similarity weight at pixel p is calculated with embedded features of the current frame $f_i^e = \mathcal{E}(f_i)$ and the warped features $f_{j\to i}^e = \mathcal{E}(f_{j\to i})$.

$$w_{j\to i}(p) = \exp\Big(\frac{f_{j\to i}^e(p)\cdot f_i^e(p)}{|f_{j\to i}^e(p)|\cdot|f_i^e(p)|}\Big) \tag{6}$$

Unlike [26], a flow field uncertainty weight is multiplied by the cosine similarity. The total weight for the reference frame j to the current frame i is as follows:

$$W_{j\to i} = w_{j\to i}\cdot\sigma(-\Sigma) \tag{7}$$

where σ is sigmoid function $\sigma(x) = 1/(1+\exp(-x))$ and $w_{j\to i}$ is cosine similarity weight. To calculate the weight, x and y-direction variances must be combined into a value

$$\Sigma = \sqrt{\Sigma_x^2 + \Sigma_y^2} \tag{8}$$

Then the aggregated feature is calculated as

$$\bar{f_i} = \sum_{j=i-K}^{i+K} W_{j\to i} f_{j\to i} \tag{9}$$

Finally, the aggregated feature is fed to the detection network and the detection results (i.e., object class and bounding box) are produced on the current frame i. With the uncertainty weight, the aggregated feature lowers the contribution of two types of features: (1) features that are far different from the feature of the current frame and (2) features that have high uncertainty.

4 EXPERIMENTS

4.1 Implementation Details

Dataset. FlyingChairs [8] is a commonly used dataset for optical flow estimation. The dataset consists of 22,872 pairs of consecutive images and the GT flow. We randomly selected 22,232 pairs for the training set and 640 pairs for the test set. We train and evaluate FlowNetU-based detector on the ImageNet dataset. ImageNet VID [20] is a large scale video object detection dataset. The ImageNet VID dataset contains 3,862 10-frame video snippets for training and 555 video snippets for evaluation. It contains 30 object categories which is a subset of 200 categories in ImageNet DET.

Fig. 3. Examples of (a), (b) input images, (c) predicted flows, (d) predicted variances of FlowNetU on Sintel Clean.

Training and Testing. For training FlowNetU, we only use FlowNetS architecture as the base model owing to its faster training and inference speed. As shown in Table 1, FlowNetS is about twice faster than FlowNetC in runtime.

We train FlowNetS and FlowNetU on the FlyingChairs dataset. Each image pair is randomly cropped to a resolution of 320×448 and flipped horizontally and vertically with probability $p = 0.5$. We use 64 as the batch size per GPU on an NVIDIA RTX 3090×2 environment. Starting from the learning rate 1e-4, we divide it by 2 at [50K, 65K, 100K, 125K] iterations with Adam optimizer [15]. No fine-tuning was applied.

Table 1. Per-frame runtime on NVIDIA RTX Titan GPU for each methods. Result of [8].

Method	Time (s)
FlowNetS	0.08
FlowNetC	0.15

In the VOD stage, the detector is trained using the pretrained FlowNetS and FlowNetU. We used the network architecture and the two-phase learning scheme of FGFA [26]. In the first phase, a single image detector and a feature extractor are trained on ImageNet DET. In the second phase, the whole networks including the flow network are trained on ImageNet VID. In the SGD training, the networks are trained for 120k and 60k iterations after 500 warm-up iterations. The initial learning rate was set to 1e-3, and the learning rate decays to 1e-4 at 80k, 40k in the first and the second phase, respectively. Data augmentation is performed during training including geometric transforms: resizing and random horizontal flip with probability $p = 0.5$. For resizing, the size of the smallest and the largest side of input images are resized to each 800 and 1333, respectively. We conducted the experiments using an NVIDIA V100 $\times$ 4 and one batch per GPU.

4.2 Evaluation of Uncertainty Measure

We visualize the uncertainty using the Sintel benchmark. Figure 3 shows scene examples from the Sintel dataset. The first two columns are input images. The third column is the predicted flows and the fourth column is the predicted variances of corresponding flows. The variance increases from blue to red.

For quantitative evaluation, we use commonly-used sparsification plots to assess the quality of the uncertainty measure [16, 23]. This metric compares two curves: the error curve and the Oracle curve which denotes the true error

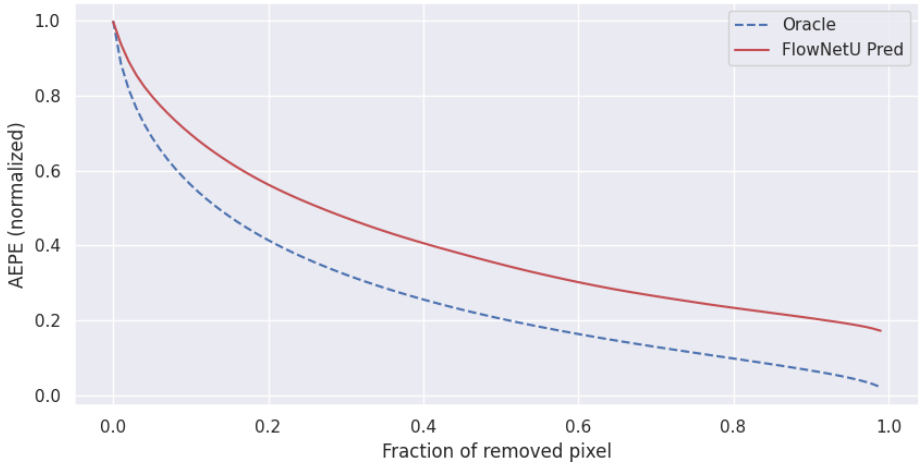

Fig. 4. Sparsification plot on the clear Sintel train-set. The Oracle curve indicates the best ranking of uncertainty.

Table 2. Area Under Curve (AUC) of error curve on Sintel and Middlebury benchmarks.

Uncertainty measure	Sintel AUC	Middlebury AUC
Gradient [2]	1.022	0.971
Energy [4]	0.470	0.498
Learned [16]	0.474	0.496
ProbClassicA [23]	-	**0.466**
Proposed	**0.415**	0.561
Oracle	0.223	0.297

curve. The error curve is plotted with every removal of a fraction of pixels in a descending order of the estimated variance. The AEPE values of the remaining pixels are normalized. Similarly, the true error curve is drawn in order of the error between the prediction and the GT. If the estimated variance is close to the actual uncertainty, the error curve should be close to the Oracle with a low area under curve (AUC) value. The Oracle provides the best order of uncertainties under the assumption that a higher uncertainty causes a higher error. Figure 4 shows the plot of our method. Table 2 shows the performance of the uncertainty measure of various methods from [2, 16, 23]. We use the results of a gradient-based method [2], an energy-based one [4] and a supervised learning [16] a confidence-measure one from [23]. These results demonstrate that our uncertainty measure works well in detecting reliable flow estimates.

4.3 Experiments on ImageNet VID

We used FlowNetS-FGFA (FlowNetS-FGFA) as the baseline method. To demonstrate the effectiveness of the uncertainty in flow field in detection, we compare the performance of our method (FlowNetU-FGFA) with the baseline. For a fair comparison, FlowNetS-FGFA and FlowNetU-FGFA are trained and evaluated using the same scheme. The result of FlowNetS-FGFA is different from that of [26] because we use Faster-RCNN [18] instead of R-FCN [7] as a single image detector. As analyzed in [26], the GT objects are categorized into $fast$, *medium* and *slow* motions according to the speed of moving objects. The speed of a moving object was calculated by intersection over union (IoU) of the box of the

objects in the nearby frames. In [26], the IoU is dubbed as *motion IoU*. The GT object is classified as $fast$ (motion IoU < 0.7), $medium$ (0.7 ≤ motion IoU ≤ 0.9) and $slow$ (motion IoU > 0.9). The mAP is widely used in object detection tasks

Table 3. Accuracy comparison of our method with the baseline on ImageNet VID. ResNet-101 [10] is used as the feature network for both methods.

	FlowNetS-FGFA	FlowNetU-FGFA (proposed)
mAP (%)	75.79	77.06
mAP (%) ($fast$)	51.54	52.11
mAP (%) ($medium$)	74.41	75.36
mAP (%) ($slow$)	83.27	84.93

as a performance metric. To compute the mAP, the IoU threshold is set to 0.5. Table 3 summarizes the performance of the proposed method and the baseline on ImageNet VID. The mAP of the proposed FlowNetU-FGFA improves by 1.27 compared to FlowNetS-FGFA. In addition, the mAP increases by 0.57, 0.95, 1.66 for $fast$, $medium$ and $slow$ motion, respectively. Table 4 shows numerical evaluation of the true positive (TP) and the FP of the baseline and those of

Table 4. Comparison of baseline and our approach for the number of TPs and FPs on ImageNet VID.

	FlowNetS-FGFA	FlowNetU-FGFA (proposed)	Variation rate (%)
TP	197,260	196,357	-0.46
FP	71,968	64,347	-10.59

FlowNetU-FGFA detections. The number of TP's and FP's was calculated with the settings of the IoU threshold of 0.5 and the score threshold of 0.5. Predicted boxes with scores lower than the threshold are considered as negative. For ImageNet VID, FlowNet-FGFA reduces both TP and FP. FlowNetU-FGFA decreases the TP rate by 0.46% and the FP one by 10.59%. Unlike the original implementation, FlowNetU-FGFA warps uncertain flow fields by small weights and results in a decrease in the TP rate. However, the decreasing rate of the FP is much larger than that of the TP.

5 CONCLUSION

In this paper, we proposed a method to improve the accuracy of video object detection and reduce the FP rate by considering the uncertainty of optical flow. We modified the FlowNet architecture to learn variances through Gaussian modeling of flow fields. Compared to the baseline method, the proposed method increased the mAP by 1.27 and reduced the FP rate by 10.59%. These results demonstrate that suppressing unreliable flow field using the uncertainty weight increases the performance and reduces the false detection considerably.

ACKNOWLEDGMENTS

This work was supported by Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2021-0-00131, Development of Intelligent Edge Computing Semiconductor For Lightweight Manufacturing Inspection Equipment)

REFERENCES

[1] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. 2011. A database and evaluation methodology for optical flow. *International journal of computer vision* 92, 1 (2011), 1–31.

[2] John L Barron, David J Fleet, and Steven S Beauchemin. 1994. Performance of optical flow techniques. *International journal of computer vision* 12, 1 (1994), 43–77.

[3] Thomas Brox, Christoph Bregler, and Jitendra Malik. 2009. Large displacement optical flow. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 41–48.

[4] Andrés Bruhn and Joachim Weickert. 2006. A confidence measure for variational optic flow methods. In *Geometric Properties for Incomplete Data*. Springer, 283–298.

[5] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. 2012. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*. Springer, 611–625.

[6] Jiwoong Choi, Dayoung Chun, Hyun Kim, and Hyuk-Jae Lee. 2019. Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 502–511.

[7] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. 2016. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*. 379–387.

[8] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. 2015. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 2758–2766.

[9] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*. PMLR, 1050–1059.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[11] Yihui He, Chenchen Zhu, Jianren Wang, Marios Savvides, and Xiangyu Zhang. 2019. Bounding box regression with uncertainty for accurate object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2888–2897.

[12] Berthold KP Horn and Brian G Schunck. 1981. Determining optical flow. *Artificial intelligence* 17, 1-3 (1981), 185–203.

[13] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. 2017. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2462–2470.

[14] Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977* (2017).

[15] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[16] Oisin Mac Aodha, Ahmad Humayun, Marc Pollefeys, and Gabriel J Brostow. 2012. Learning a confidence measure for optical flow. *IEEE transactions on pattern analysis and machine intelligence* 35, 5 (2012), 1107–1120.

[17] Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* 28 (2015), 91–99.

[19] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. 2015. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1164–1172.

[20] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision* 115, 3 (2015), 211–252.

[21] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. 2018. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 8934–8943.

[22] Shiyao Wang, Yucong Zhou, Junjie Yan, and Zhidong Deng. 2018. Fully motion-aware network for video object detection. In *Proceedings of the European conference on computer vision (ECCV)*. 542–557.

[23] Anne S Wannenwetsch, Margret Keuper, and Stefan Roth. 2017. Probflow: Joint optical flow and uncertainty estimation. In *Proceedings of the IEEE International Conference on Computer Vision*. 1173–1182.

[24] Junqing Wei, Jarrod M Snider, Junsung Kim, John M Dolan, Raj Rajkumar, and Bakhtiar Litkouhi. 2013. Towards a viable autonomous driving research platform. In *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 763–770.

[25] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. 2013. DeepFlow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*. 1385–1392.

[26] Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. 2017. Flow-guided feature aggregation for video object detection. In *Proceedings of the IEEE International Conference on Computer Vision*. 408–417.

[27] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. 2017. Deep feature flow for video recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2349–2358.