UNSUPERVISED MOTION FLOW ESTIMATION BY GENERATIVE ADVERSARIAL NETWORKS

Stefano Alletto & Luca Rigazio
Panasonic Silicon Valley Laboratory
10900 North Tantau Avenue, Suite 200
Cupertino, CA 95014, USA
{name.surname}@us.panasonic.com

ABSTRACT

In this paper we address the challenging problem of unsupervised motion flow estimation. Under the assumption that image reconstruction is a super-set of the motion flow estimation problem, we train a convolutional neural network to interpolate adjacent video frames and then compute the motion flow via region-based sensitivity analysis by backpropagation. We postulate that better interpolations should result in better motion flow estimation. We then leverage the modeling power of energy-based generative adversarial networks (EbGAN's) to improve interpolations over standard L2 loss. Preliminary experiments on the KITTI database confirm that better interpolations from EbGAN's significantly improve motion flow estimation compared to both hand-crafted features and deep networks relying on standard losses such as L2.

1 INTRODUCTION

Accurate motion flow estimation is a key component of autonomous systems vision pipelines. Currently the large-scale, depth-labeled datasets needed to develop robust systems are lacking (Geiger et al. (2013); N.Mayer et al. (2016)). We address this issue with a novel, unsupervised motion-flow estimation algorithm that exploits the modeling power of generative adversarial networks. We infer region-wise matches between adjacent frames and subsequently estimate the motion flow, with no supervision or depth ground truth.

Like in the work of Long et al. (2016), we postulate that learning to interpolate adjacent video frames will drive a representation that associates related pixels, thus inherently solving for the motion flow estimation problem. Specifically, given three consecutive frames $(f1, f2, f3)$, we train a deep architecture that learns to interpolate $f2$ from $f1$ and $f3$. Then, by backpropagating output regions through the network we obtain sensitivity maps representing accurate motion flow for the scene. Since the network is only trained to solve for interpolation, this effectively removes the need of a large-scale dataset providing ground-truth flow maps.

Under this framework, improving interpolation quality would then improve motion flow estimation; this motivated us to focus on EbGAN's for our architecture, since they recently provided state of the art results in several challenging generative and auto-encoding settings.

2 BACKGROUND

Generative adversarial networks (Goodfellow et al. (2014)) employ two separate components, a *generator G* network and a *discriminator D*. The objective of the discriminator is to maximize the probability of correctly discerning between samples from a true data distribution and samples generated by the generator network. Conversely, the generator is trained with the goal of maximizing the probability of its samples being classified from the true data distribution by the discriminator. More formally, this translates into a minmax game where D maximizes:

$$L_D = log(D(x)) + log(1 - D(G(z))) \tag{1}$$

and G minimizes:

$$L_G = log(1 - G(z))) \tag{2}$$

where x and z are respectively a sample from the true data distribution and a sample from a random noise distribution. In the case of the generating images, adopting an adversarial loss leads to sharper and higher quality outputs compared to standard losses such as L1 or L2 (Isola et al. (2016); Goodfellow et al. (2014)). Recently, the EbGAN (Zhao et al. (2016)) model has been proposed: it improves on the original GAN by replacing the discriminator with an auto-encoder and outputting a pixel-level energy map instead of a binary value. During training, EbGAN's reconstruction loss results in different gradients directions inside a minibatch, leading to more efficient training and higher batch-sizes without efficiency loss.

3 UNSUPERVISED MOTION FLOW ESTIMATION

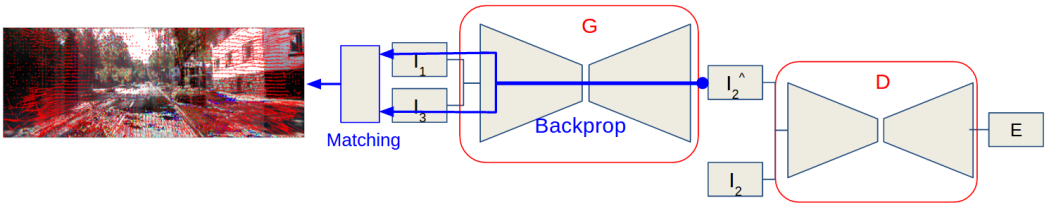

Figure 1: The adopted GAN architecture

In order to estimate the motion flow between two consecutive frames in an unsupervised fashion, we follow the framework proposed by Long et al. (2016). Given two frames at time t and $t + 2$, the motion flow can be estimated by training a network to interpolate the frame at $t + 1$ and performing per-pixel backpropagation through the network itself. For each pixel in $f2$, this results in two sensitivity maps, one for $f1$ and one for $f3$. Sampling from these maps allows us to find pixel-level correspondences between $f1$ and $f3$ and hence the motion flow.

Intuitively, the quality of the motion flow obtained by backpropagating individual pixels of $f2$ is related to the quality of the interpolation. To increase the quality of such interpolation, we devise an *energy based generative adversarial network* (EbGAN) Zhao et al. (2016) where the generator G is tasked with the interpolation of the two input frames and D outputs a pixel-level energy map instead of functioning as a binary classifier between *true* or *fake*. The overall architecture of the network, shown in Figure 3, consists in two fully convolutional auto-encoders, G and D. Following recent works in motion flow estimation (Zhu et al. (2017); Yu et al. (2016); Long et al. (2016); Ng et al. (2016)), we use the *FlowNet Simple* (Fischer et al. (2015)) architecture in both G and D, and reduce the number of conv-deconv blocks in D from 5 to 4 and remove skip connections. We use *leaky relu* activations except for the output layer which employs a *tanh* activation. Furthermore, *virtual batch normalization* (Salimans et al. (2016)) is introduced after every block of G.

We combine the original EbGAN loss with a *content loss term* in the generator loss:

$$L_D((f1, f3), f2) = D(f2) + max(0, m - D(G(f1, f3))) \tag{3}$$

$$L_G(f1, f3) = \lambda D(G(f1, f3)) + (1 - \lambda) MSE(f2, G(f1, f3)) \tag{4}$$

where m is a positive margin, $\lambda \in [0, 1]$ is a positive constant and MSE is the standard mean squared error function. The hyperparameter λ in Eq. 4 controls the balance between adversarial and content losses, and is very important during the first few epochs, where the discriminator is likely to prevail. We empirically set λ to 0.1 by cross-validation.

We train the network using the Adam optimizer (Kingma & Ba (2014)) with β_1 set to 0.5 and minibatch size of 16, 1000 batches per epoch and the training is stopped after 250 epochs. Once the training is completed, to obtain the motion flow we disregard D and backpropagate only through G. Also, we don't perform pixel backpropagation like in (Long et al. (2016)), instead we enforce local smoothness in the sensitivity maps by backpropagating small regions (e.g. 4×4 pixels instead of individual pixels with stride). With 128×384 images, backpropagating a 4×4 region takes 70 ms

Table 1: Performance comparison on the KITTI FLOW 2012 training set

Method	Acc@5	APE
HoG	0.455	9.68
KLT	0.702	8.16
L2	0.640	5.967
Yu et al. (2016)	-	4.3
Adversarial	0.710	4.89

on a TITAN-X GPU, resulting in a total time 215 seconds per image with stride of 4. This process will produce two sensitivity maps $S_{\tau,i,j}$ for the region with center position at coordinates $(i,j)_{f2}$ for frame $\tau \in [f1, f3]$. We then compute the correspondence between the projection of $(i,j)_{f2}$ in each frame τ by selecting the $\underset{\tau}{\text{argmax}} S_{\tau,i,j}$. Subsequently, the matching between the regions in $f1$ and $f3$ is obtained exploiting the transitive relation between $f1, f2$ and $f2, f3$.

4 Experimental Evaluation

4.1 Dataset and metrics

We train the network in an unsupervised manner using the KITTI RAW database (Geiger et al. (2013)), containing 159 video sequences acquired from a car-mounted camera with different objects with various motion patterns. We do not perform any data augmentation. To assess the motion flow accuracy we use ground-truth from the KITTI FLOW 2012 dataset training set, containing 194 frame pairs with ground truth motion flow. We provide evaluation metrics widely used for this settings: Accuracy@k, meaning the ratio of motion vectors with end point error lower than k pixels, and APE which is the average point error of all motion vectors.

4.2 Preliminary evaluation

We evaluate our algorithm against two popular hand-crafted methods, KLT (Tomasi & Kanade (1991)) and HoG (Brox & Malik (2011)), and against a baseline unsupervised algorithm where the generator is trained using the L2 loss. Notably, we keep the same architecture for both adversarial and L2 training to evaluate the impact of the adversarial loss motion flow estimation. Moreover, to provide a complete picture of the current state of the art we include results from Yu et al. (2016) which, to the best of our knowledge, obtains the best results for unsupervised flow estimation.

Preliminary results are reported in Table 1: we remark that using the adversarial training results in a significant improvement of 7% accuracy@5 and 1 pixel in APE over the L2 baseline[1]. A further comparison with methods relying on hand-crafted features, namely KLT and HoG, confirms the superiority of the representation learned by our network. Finally, it is worth noting that the method by Yu et al. (2016) heavily relies on both geometrical and photometrical data augmentation, as long as multiple datasets; conversely our results are obtained by directly training on the KITTI RAW dataset, which includes less than 50,000 training images [2].

5 Conclusions

Motion flow estimation is a key component of both human and robotics vision pipelines. Today motion flow is computed either with hand-crafted or by deep architectures trained in a supervised manner. We believe that unsupervised motion flow estimation is very desirable and within reach, and can unlock the potential of large video datasets to provide very robust performance. Preliminary results show performance superior to hand-crafted methods and close to state of the art for unsupervised methods. One issue with the backpropagation approach is that it is still slow, motivating us to look next at hybrid forward/backward architectures.

[1] We also implemented the architecture from Long et al. (2016), however despite using the same data augmentation described in the paper, we were unable to reproduce the reported results

[2] Source code and pre-trained models will be released to foster future research on the topic

REFERENCES

Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE transactions on pattern analysis and machine intelligence*, 33(3): 500–513, 2011.

Philipp Fischer, Alexey Dosovitskiy, Eddy Ilg, Philip Häusser, Caner Hazırbaş, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. *arXiv preprint arXiv:1504.06852*, 2015.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Gucan Long, Laurent Kneip, Jose M Alvarez, Hongdong Li, Xiaohu Zhang, and Qifeng Yu. Learning image matching by simply watching video. In *European Conference on Computer Vision*, pp. 434–450. Springer, 2016.

Joe Yue-Hei Ng, Jonghyun Choi, Jan Neumann, and Larry S Davis. Actionflownet: Learning motion representation for action recognition. *arXiv preprint arXiv:1612.03052*, 2016.

N.Mayer, E.Ilg, P.Häusser, P.Fischer, D.Cremers, A.Dosovitskiy, and T.Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pp. 2226–2234, 2016.

Carlo Tomasi and Takeo Kanade. Detection and tracking of point features. 1991.

Jason J Yu, Adam W Harley, and Konstantinos G Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. *arXiv preprint arXiv:1608.05842*, 2016.

Junbo Zhao, Michael Mathieu, and Yann LeCun. Energy-based generative adversarial network. *arXiv preprint arXiv:1609.03126*, 2016.

Yi Zhu, Zhenzhong Lan, Shawn Newsam, and Alexander G Hauptmann. Guided optical flow learning. *arXiv preprint arXiv:1702.02295*, 2017.