MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Learning Occlusion-Aware Dense Correspondences for Multi-Modal Images

Shimoya, Ryosuke; Morimoto, Tahashi; van Baar, Jeroen; Boufounos, Petros T.; Ma, Yanting; Mansour, Hassan

TR2022-149 December 01, 2022

Abstract

We introduce a scalable multi-modal approach to learn dense, i.e., pixel-level, correspondences and occlusion maps, between images in a video sequence. The problems of finding dense correspondences and occlusion maps are fundamental in computer vision. In this work we jointly train a deep network to tackle both, with a shared feature extraction stage. We use depth and color images with ground truth optical flow and occlusion maps to train the network end-to- end. From the multi-modal input, the network learns to estimate occlusion maps, optical flows, and a correspondence embedding providing a meaningful latent feature space. We evaluate the performance on a dataset of images derived from synthetic characters, and perform a thorough ablation study to demonstrate that the proposed components of our architecture combine to achieve the lowest correspondence error. The scalability of our proposed method comes from the ability to incorporate additional modalities, e.g., infrared images.

IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) 2022

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Learning Occlusion-Aware Dense Correspondences for Multi-Modal Images

Ryosuke Shimoya[1]
Mitsubishi Electric Corporation
Yokohama, Kanagawa, Japan
Shimoya.Ryosuke@da.MitsubishiElectric.co.jp

Takashi Morimoto[1]
Woven Core, Inc.
Tokyo, Japan
takashi.morimoto@woven-planet.global

Jeroen van Baar[1]
Amazon Robotics
North Reading, MA
jeroenvb@amazon.com

Petros Boufounos, Yanting Ma, Hassan Mansour
Mitsubishi Electric Research Laboratories
Cambridge, MA
{petrosb, yma, mansour}@merl.com

Abstract

We introduce a scalable multi-modal approach to learn dense, i.e., pixel-level, correspondences and occlusion maps, between images in a video sequence. The problems of finding dense correspondences and occlusion maps are fundamental in computer vision. In this work we jointly train a deep network to tackle both, with a shared feature extraction stage. We use depth and color images with ground truth optical flow and occlusion maps to train the network end-to-end. From the multi-modal input, the network learns to estimate occlusion maps, optical flows, and a correspondence embedding providing a meaningful latent feature space. We evaluate the performance on a dataset of images derived from synthetic characters, and perform a thorough ablation study to demonstrate that the proposed components of our architecture combine to achieve the lowest correspondence error. The scalability of our proposed method comes from the ability to incorporate additional modalities, e.g., infrared images.

1. Introduction

Tracking of humans in a video sequence is a fundamental problem in surveillance applications. Especially in multimodal applications, tracking in one modality can assist imaging in another. For example, in combined optical and radar applications, tracking is easier in the optical domain, whereas detection of occluded objects is easier in the radar domain. In this paper we focus on the tracking component of such systems, providing a method to determine both dense correspondences between video frames (i.e., pixel-to-pixel correspondences), and occlusion maps which indicate which parts (i.e., pixels) of one frame are not visible in the next and which parts in the next frame are new and do not appear in the first.

Determining dense, i.e., pixel-to-pixel, correspondences between images in a video sequence is a fundamental problem in computer vision. We are particularly interested in dense correspondence between frames of walking people. Most "classical" people tracking methods are sparse, either tracking a bounding box or parts/skeletons for pose tracking, e.g. [12, 11]. Other human tracking methods have been introduced to recover correspondences, such as random forests for volumetric pose tracking [2], but these tracking methods do not estimate dense pixel-to-pixel correspondences. Instead, optical flow (OF) is a well-studied alternative that computes a dense pixel flow between pairs of successive frames; see [1] for a comprehensive list of references.

In addition, we are interested in detecting which parts in one frame are occluded in the next, and vice versa. As a person is walking in front of an optical sensor, different parts of the body are occluded at different frames, as each part has a separate motion. For example, an arm could be occluding part of the torso in one frame but not in another. While, in principle, it is possible to detect such occlusions using only accurate optical flow or dense correspondences, we found that explicit detection provides better results.

Deep learning has significantly improved the state-of-the-art for dense correspondences. In particular, recent deep learning based OF approaches can estimate flow even in the presence of larger displacements [3, 6]. Other deep learning approaches estimate dense correspondences as a means to ultimately recover 3D human pose [5, 14]. The dense correspondences are estimates of so-called *uv*-maps which have a direct relation to 3D human shape parts. The emphasis is on recovering accurate 3D human pose, rather than

[1]RS, TM and JB performed this work while at MERL

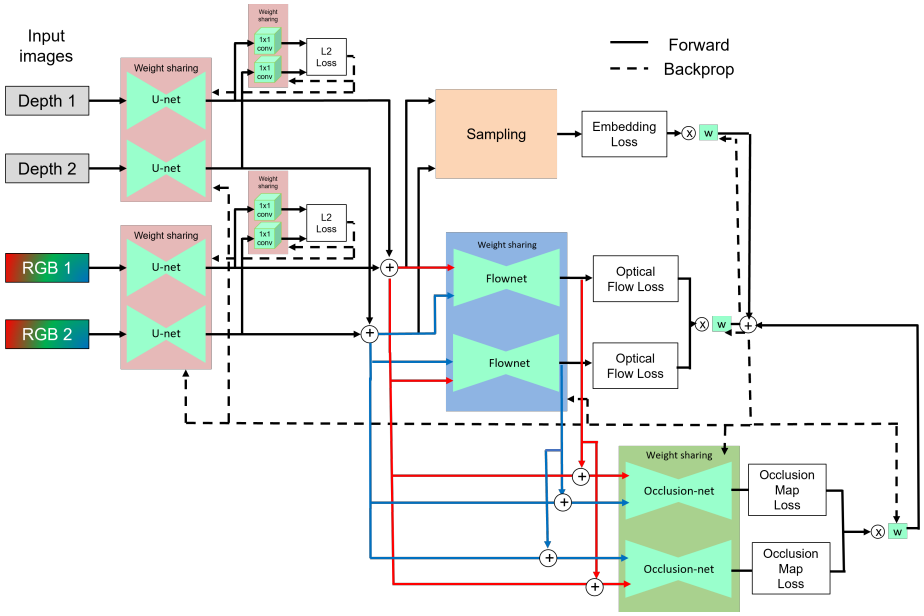

Figure 1. Our proposed scalable multi-modal network architecture. The RGB and depth images are encoded to per-pixel feature vectors, with a separate encoder for each modality. The per-pixel features are concatenated and input to a correspondence embedding, optical flow, and occlusion map estimation. We add supervision on reconstruction to force auto-encoders to learn features representing the whole image.

accurate dense correspondences. The method proposed by Wei et al. [13] on the other hand, aims to estimate dense correspondences using depth images as input. The correspondence problem is formulated as a many-class classification problem. A water-tight mesh is segmented into many small regions and training aims to learn an embedding of how depth pixels relate to neighboring depth pixels through the segmentation and classification. The method has drawbacks: the need for water-tight meshes, and training that is both multi-pass and computationally very expensive.

The methods above can handle challenging pose differences between frames, but sacrifice accuracy, particularly at the extremities such as arms and hands. Lack of accuracy is especially prominent when tracking walking people. Our work aims to address this by a) exploiting multi-modal input, e.g., depth and color, and b) augmenting the optical flow estimation with an additional correspondence embedding using a contrastive learning approach. The proposed architecture (Fig. 1) consists of auto-encoders to extract meaningful per-pixel features. The per-pixel features are concatenated to learn three tasks: multi-modal optical flow, occlusion detection, and embedding via contrastive learning. Our contributions are summarized as follows:

- A scalable multi-modal network architecture, combining optical flow and occlusion estimation with a contrastive based correspondence embedding.
- A dense correspondence inference method which takes both latent space distance, as well as Euclidean distance into account.
- A thorough evaluation of performance on synthetic ground truth data.

Using multiple modalities is not unique in our approach nor is it constraining. Since RGBD sensors are becoming inexpensive and ubiquitous, many methods propose to exploit both depth and color [4, 10]. The additional information in the depth dimension provides robust information that significantly improves the performance of such methods. Furthermore, optical flow estimates are often used for learning tasks that involve temporal image data, e.g., [8].

The next section introduces our approach, including the network architecture, the training method and the inference process. Section 3 provides experimental results and ablation studies that validate our approach, and Section 4 discusses our results and concludes.

2. Network Design

To produce the dense correspondences and the occlusion maps we propose a scalable deep neural network architecture that is trained end-to-end using multi-modal image pairs as input. Currently, the multi-modal input consists of pairs of both RGB and depth images.

The depth and color input images are first encoded using an auto-encoder—a U-net [9] in our case. The auto-encoder learns per-pixel image structure represented by an m_{Mod}-dimensional latent space, where Mod refers to the modality

of the input, i.e., RGB or depth. We train separate U-nets for depth and for color. The weights for the U-nets are shared between the pairs of input, see Fig. 1. Additional supervision to reconstruct the original input images is provided by 1×1 convolutions after the encoding.

The per-pixel m_{Mod}-dimensional feature vectors from the depth and color images are concatenated into a final feature vector of dimension $m_d + m_c$ and input to three trainable tasks: optical flow estimation, correspondence embedding via sampling, and occlusion detection.

Both optical flow estimation and occlusion detection are networks based on the FlowNet architecture [6]. The only difference is that the occlusion detection network, which we call OcclusionNet, outputs a scalar output in $[0, 1]$ for every pixel indicating occlusions, whereas FlowNet outputs a 2D vector output estimating the flow for every pixel of the input. Furthermore, we found that including the FlowNet output in the OcclusionNet, by concatenating it with the feature vectors from the U-nets improves the OcclusionNet results.

During training, we use both directions for flow and occlusion detection (i.e., from the first to the second frame and from the second to the first), thus providing additional training data to train the network. To do so, we use the same FlowNet and OcclusionNet twice, with reversed inputs and shared weights, as indicated in Fig. 1. If an application only requires one-directional flow or occlusion estimation, the branches estimating the reverse direction can be eliminated during inference.

2.1. Losses

The loss function used for training comprises four kinds of losses: a reconstruction loss in the output of the U-nets, that forces autoencoders to represent the whole input, an embedding correspondence loss, that promotes embeddings that correctly represent correspondences between points in the frames, an optical flow loss, which promotes models that accurately learn the optical flow, and an occlusion estimation loss, which promotes models that accurately estimate occlusions.

The reconstruction loss L_r is a mean squared error loss between the original and reconstructed image, and is computed separately for the color (L_r^c) and the depth image (L_r^d). The correspondence embedding loss is contrastive, of the form:

$$\begin{aligned} L_e = & \sum_{i=1}^{N} y_i \|D(p_{1,i}) - D(p_{2,i})\|_2^2 \\ & + (1 - y_i)\max(0, C - \|D(p_{1,i}) - D(p_{2,i})\|_2^2) \\ \text{s.t.} & \begin{cases} y_i = 1 & \text{if } p_{1,i} \iff p_{2,i} \\ y_i = 0 & \text{otherwise} \end{cases} \end{aligned} \quad (1)$$

The functions $D(\cdot)$ represent the encoding using U-nets and subsequent concatenation of latent vectors. If the i^{th} pixel pair $(p_{1,i}, p_{2,i})$ between the images 1 and 2 is in correspondence, we set its corresponding label to $y_i = 1$, and $y_i = 0$ otherwise. In (1), C denotes the contrastive margin, a hyper-parameter to help stabilize the learning. Thus, (1) aims to promote similar feature vectors in each image for pixels in correspondence, and dissimilar feature vectors for ones not in correspondence.

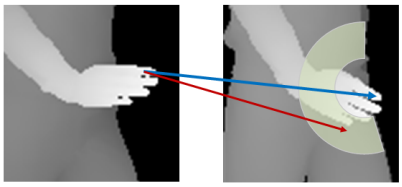

Figure 2. Sampling a non-corresponding pixel. The blue arrow denotes the correspondence based on the ground truth optical flow. A non-corresponding pixel is selected within the range $[min\_dist, max\_dist]$

During training, we sample a total of N pixel pairs: N_{co} in *known* correspondence, and $N_{nc} = N - N_{co}$ in *known* non-correspondence. The pairs are sampled randomly, but we adopt the following sampling strategy:

- avoid sampling too close to previously sampled pairs to reduce bias, and
- for the non-corresponding samples, sample at least min_dist away from the known correspondence, and at most max_dist.

The strategy is illustrated in Fig. 2. Both min_dist and max_dist are hyper-parameters. We want to simultaneously promote smoothness in the dissimilarity and only consider non-corresponding points that would be meaningful by being sufficiently close to corresponding ones. Allowing non-corresponding points too far away would contribute little to the learning.

Since our optical flow network is based on FlowNet [3] we employ the same end-point L_2 loss as FlowNet. Since occlusion detection is a binary-valued detection problem, we use the entropic loss at its output.

The total loss is therefore composed as a weighted sum:

$$L = w_e L_e + w_r L_r^d + w_r L_r^c + w_{of} L_{of} + w_{oc} L_{oc}, \quad (2)$$

with weights w for each loss.

2.1.1 Multi-Task Losses

The authors in [7] recognized that treating the weights in (2) as fixed hyper-parameters may hinder learning. They propose a multi-task loss approach where the weights are simultaneously learned during training. We adopt their approach and formulate the loss as:

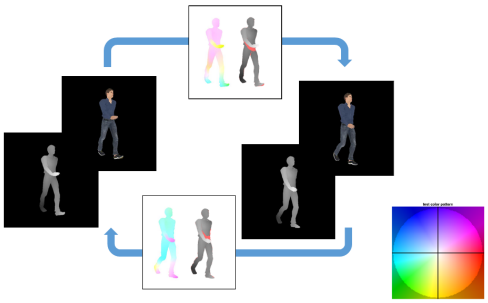

Figure 3. An example pair of input data. In addition to color and depth, we provide optical flow and occlusion maps between the image pairs in both directions. The color wheel insert indicates direction and magnitude for the optical flow.

$$\begin{aligned} L =& L_e \cdot \exp(-\log \sigma_1^2) + L_r^d \cdot \exp(-\log \sigma_2^2) \\ &+ L_r^c \cdot \exp(-\log \sigma_3^2) + L_{of} \cdot \exp(-\log \sigma_4^2) \\ &+ L_{oc} \cdot \exp(-\log \sigma_5^2) + \sum_{i=1}^{5} \log \sigma_i^2. \end{aligned} \tag{3}$$

The parameters σ_i in (3) require a reasonable initial guess.

2.2. Training

To train the network, we input depth and RGB image pairs of dimension 512×512. The U-net autoencoders output is $512 \times 512 \times 32$, and we stack those into $64-$dimensional per-pixel features.

Our training data requires ground-truth optical flow and occlusion maps in both directions per image pair, corresponding to input RGB and depth images. To obtain sufficient training examples, we use synthetic training data. The optical flow, occlusion maps, color and depth images are used for supervision in the training. We have generated 38 synthetic characters with a variety of clothing appearances. We hold out data associated with two synthetic characters for testing. Fig. 3 depicts data for an example image pair.

2.3. Flow Inference

Inference of dense correspondences combines information from both optical flow estimation and the features computed by the U-nets. Combining this information allows us to improve the outcome, compared to only using one of the two sources of information.

The features computed by the U-nets are used through the Feature Distance FD between pixel i in frame 1, and pixel j in frame 2. Given two sets of pixels $\mathcal{P}_1, \mathcal{P}_2$, corresponding to the two frames, the Feature Distance $FD_{(i,j)}$ between $i \in \mathcal{P}_1$ and $j \in \mathcal{P}_2$ is defined as

$$FD_{(i,j)} = \|D(p_{1,i}) - D(p_{2,j})\|_2^2. \tag{4}$$

Absent optical flow information, the corresponding pixel to a pixel $i \in \mathcal{P}_1$ in frame 1 is the closest pixel j in frame 2 with respect to the feature distance, i.e., $\arg\min_{j \in \mathcal{P}_2} FD_{i,j}$.

This estimate of corresponding points does not impose any local smoothness constraints. Thus, outlier matches may occur, in which pixels are selected as corresponding even though they are far away from nearby corresponding ones. Thus, incorrect correspondences may be selected. This is the main source of errors using this approach.

The optical flow estimate $\widehat{OF}$ directly provides a high-quality estimate of correspondences. Compared to the estimation described above, this estimate is much smoother and makes different kind of errors which are in the vicinity of the ground truth and do not suffer from outliers of large magnitude. Thus, this estimate can be used as a smoothness constraint to regularize the estimate above.

To impose this smoothness constraint we penalize correspondence candidates that are far away from the estimated optical flow, $\widehat{OF}$, obtained during inference. We define

$$OD_{(i,j)} = \|(p_{1,i} + \widehat{OF}(p_{1,i})) - p_{2,j}\|_2^2, \tag{5}$$

by simply adding the estimated optical flow vector for $p_{1,i}$ to $p_{1,i}$. We can now define the combined distance

$$CD_{(i)} = \min_{j \in \mathcal{P}_2} (FD_{(i,j)} + \lambda OD_{(i,j)}), \ \ \forall i \in \mathcal{P}_1. \tag{6}$$

Furthermore, the correct distance measurement OD should take into account the three-dimensional distance of points in or (5), i.e., the distance over the volume seen by the image, rather than a simple pixel distance. To compute this distance we use the 2D pixel location is given as $p = (x, y)$ concatenated with a normalized depth value $\tilde{z}$ to define a '3D' pixel location $p = (x, y, \tilde{z})$. The normalized depth value is computed to have the same scale as the pixels

$$\tilde{z} = \frac{d_{p_i}}{d_{max} - d_{min}} * \frac{H_{pixel}}{H_{real}}, \tag{7}$$

where d_{p_i} is the depth at pixel p_i, d_{max}, d_{min} are the largest and smallest depth resp., H_{pixel} is the height of the person in pixels, and finally H_{real} is the height of the person in meters. We discuss these parameters in more detail in Sec. 3.

2.4. Occlusions

The ground truth optical flow incorporates information for the occlusion maps and can be used to derive them, even if not available from the ground truth generation. Specifically, for any two frames 1 and 2, when generating the synthetic training data we compute optical flow for both $1 \rightarrow 2$ and $2 \rightarrow 1$. A pixel at position p_1 in frame 1 that moves to position p_2 in frame 2 using the optical flow $OF_{1 \rightarrow 2}$ is visible in both frames if and only if the optical flow from frame

2 to frame 1, $OF_{2\to1}$ returns pixel p_2 in frame 2 back to p_1 in frame 1. In other words, p_1 in frame 1 is not occluded in frame 2 only if $p_2 + OF_{2\to1}(p_1 + OF_{1\to2}(p_1))$ is equal to p_1. This condition imposes *cycle*-consistency in the optical flow for pixels that are not occluded.

The cycle-consistency condition above assumes that pixels are continuous or that optical flow is discretized. In practice the flow from one frame to the other is discretized to the nearest pixel. Therefore, a tolerance factor is necessary in implementing the condition. Taking into account the rounding effects, we mark a pixel as occluded if

$$\|[p_2 + OF_{2\to1}([p_1 + OF_{1\to2}(p_1)])] - p_1\|_\infty \geq 1, \quad (8)$$

where $[\cdot]$ denotes rounding to the nearest integer. In other words, we allow 1 pixel discrepancy in either direction in the round-trip flow to account for the rounding effects. This approach allows us to generate ground truth occlusion maps for training.

Occlusion maps are also important in evaluating optical flow estimates both during training and during evaluation. In particular, true optical flow in areas where an occlusion occurs between subsequent frames is undefined. We only consider unoccluded pixels when we determine flow errors in the evaluation, as we discuss next.

3. Experiments

We implemented our network in PyTorch and performed several experiments to evaluate the performance. Our training data consists of color, depth and optical flow images related to 36 synthetic characters, for a total of 3.5k multimodal images. We train for 120 epochs and validate every epoch, to verify if any overfitting occurs. For Eq. 6 we use $\lambda = \frac{1}{c} FD_0$, where FD_0 denotes a feature match (according to Eq. 4) with smallest distance to $p_{1,i} + \widehat{OF}(p_{1,i})$. We use $c = 5$ in all our experiments. The parameters for Eq. 7 can mostly be determined from the input images. The H_{pixel} could be determined from a bounding box around the person, for example if we were to use a region proposal based object detector to detect people. The H_{real} could be determined if camera intrinsics were given in metric space, or if a known scale could be identified in the images. We instead took an average over the height of our synthetic characters and used this average in all experiments.

3.1. Flow Estimation

We first evaluated the performance on flow estimation by eliminating the OcclusionNet from the network. Our evaluation is on data for two held out synthetic characters. Table 1 shows the performance of our proposed approach. The table lists the RMS pixel error using only the estimated optical flow ($\widehat{OF}$), only the the feature descriptor distance (FD), and combined optical flow and feature distance as described in Section 2.3 (CD). The best model (row A) obtains a RMS pixel error of 1.95 for the combined feature and optical flow distance (Eq. 6). Our best model uses $C = 2.0$ (Eq. 1), $min\_dist = 5$ and $max\_dist = 50$. We performed ablation studies to understand the effect of the hyper-parameters on network performance. We found that performance is robust over a range of values for the margin C. We reduced the U-net depth from 5 to 4 (B) and 2 (C) layers resp., which clearly impacts performance. For training we use $N_{co} = 250$ and $N_{nc} = 250$, and thus $N = 500$ for Eq. 1. Reducing to $N = 50$ (D) drastically decreases performance. Increasing the number of samples beyond $N = 500$ however, showed no effect. We use the Adam optimizer for training. Imposing $L2$ regularization (E) has a surprisingly detrimental effect on performance. Omitting the supervision for reconstruction after U-net encoding (F) results in a performance loss.

	Name	$\widehat{OF}$	FD	CD
A	Best model	3.88	3.46	**1.95**
B	U-Net d=4	3.92	13.39	2.87
C	U-Net d=2	18.80	25.83	8.79
D	50 samples	4.04	17.03	2.89
E	L2 reg.	4.20	22.24	3.00
F	No rec.loss	3.96	3.97	2.98
G	Depth only	-	3.39	-
H	Color + OF	3.99	37.89	3.06
I	Color + Contr.	-	3.98	-
J	Color + OF + Contr.	3.93	3.22	2.36
K	Color + Depth + OF	4.09	24.13	3.04
L	16-dim	3.92	10.05	2.61
M	8-dim	3.91	10.89	2.59

Table 1. Results for trained network. Reported numbers represent RMS pixel error for estimated optical flow only, feature descriptor distance only, and combined optical flow and feature distance respectively. See text for details.

We further performed ablation studies by omitting parts of the network. Using only depth image and contrastive loss (G) results in a good embedding. For ablations H, I, J we omit depth input, and train with optical flow only (H), contrastive loss only (I) and both (J). It's clear that combining both losses performs best in this case as well, but not as good as our best model A. Finally, we omit contrastive loss, and train with all modalities and optical flow (K). The model clearly underperforms our best model.

We use a latent space dimension of 32 for each U-net encoding, and hence when stacking color and depth we obtain 64-dimensional per-pixel features. Results for 16 (32 stacked), and 8 (16 stacked) dimensions are shown in rows L and M. Fewer dimensions clearly impact performance for

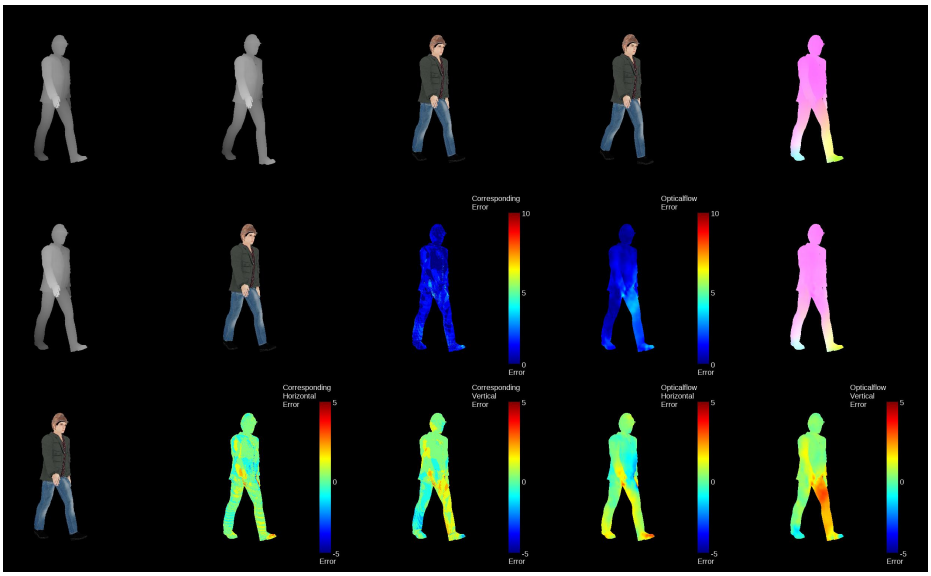

Figure 4. Qualitative example performance. Top row shows the input depth and color frame 1 and 2 images resp., and ground truth optical flow $1 \rightarrow 2$. The first two images in the middle row show reconstructed frame 1 depth and color images. The next two images show errors for Eq. 6 and optical flow only. The right-most image in middle row is the estimated optical flow $\widehat{OF}$. The bottom row left-most image is the reconstructed frame 1 color image using Eq. 6. The remaining images show horizontal and vertical errors for the error image in the middle row. As demonstrated, the error in optical flow on the character's leg is significantly reduced when combining with feature distance.

the feature distance only correspondence estimates. However, optical flow and our proposed combined inference method, Eq. 6, mitigate the decrease. However, both models clearly underperform our best model.

We should note that the ablation studies also provide an indirect comparison with the state of the art in this area. In particular, omitting optical flow for inference and instead using only the feature distance can be regarded as a comparison to the method in [13]. Similarly, using only optical flow is equivalent to [6]. Nevertheless, we should compare performance directly, which we note as future work.

Figure 4 shows a qualitative result for our proposed approach. Please refer to the figure caption for an explanation of the layout. Although the reconstructed images look visually similar, the error images reveal that the our proposed method reduces the error in the optical flow estimation, particularly evident on the character's leg.

3.2. Occlusion Estimation

Next, we incorporated the OcclusionNet in the architecture, to evaluate its effect, both on the performance of the correspondence estimation, as well as the performance of the occlusion estimation. The effect on correspondence estimation is quantified in Table 2. Row A uses the same configuration in row A of Table 1 as the baseline. Row D further incorporates the complete OcclusionNet, as illustrated in Fig. 1. In addition, we perform two ablation studies. In one we only use the features computed by the U-nets as input to the OcclusionNet, eliminating the output of the FlowNet from the OcclusionNet input (row B). In the other we only use the output of the FlownNet as input to the OcclusionNet, eliminating the features from its input (row C). Finally, we completely eliminate the sampling network from the system (light orange in Fig. 1), keeping both feature and flow inputs to the OcclusionNet (row E).

	Name	$\widehat{OF}$	FD	CD
A	No OcclusionNet	3.88	3.46	1.95
B	Features Only	3.97	3.84	1.87
C	Flow Only	4.04	6.56	2.27
D	Features + Flow	3.86	4.82	1.93
E	No sampling network	3.77	34.14	4.31

Table 2. Correspondence results for trained network with the addition of the OcclusionNet. Reported numbers represent RMS pixel error for estimated optical flow only, feature descriptor distance only, and combined optical flow and feature distance respectively. See text for details

As evident for row C in the table, eliminating the features from the OcclusionNet input has a significant degrading effect in the performance of the correspondence estimation. Furthermore, our preliminary studies showed that this configuration also performs poorly in occlusion estimation, so we decided to remove it from subsequent experiments.

We should note that, in principle, the flow estimates, if accurate, should be sufficient to estimate occlusions. Af-

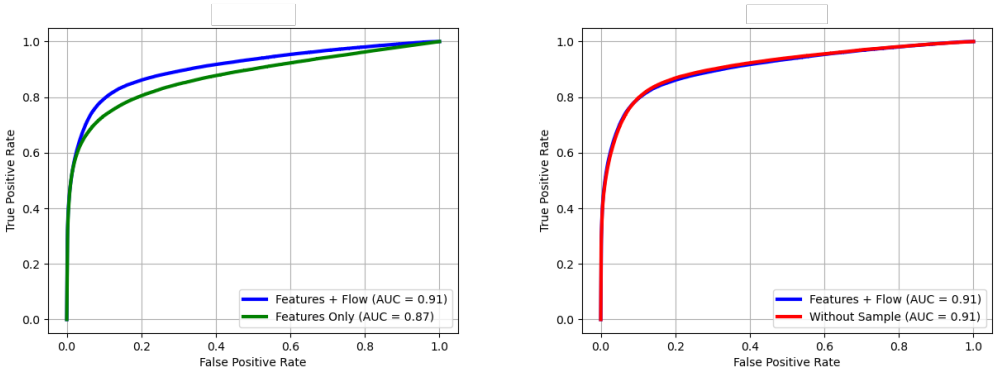

Figure 5. Receiver Operating Characteristic (ROC) for occlusion detection. Left plot is comparing OcclusionNet performance using only Features as input (Green) vs using Features concatenated with estimated optical flow (Blue). Right plot is comparing the effect of including the sampling network in the training (Blue) vs. removing it (Red). Taking into account estimated optical flow improves performance, while the presence of the sampling circuit in training makes no difference.

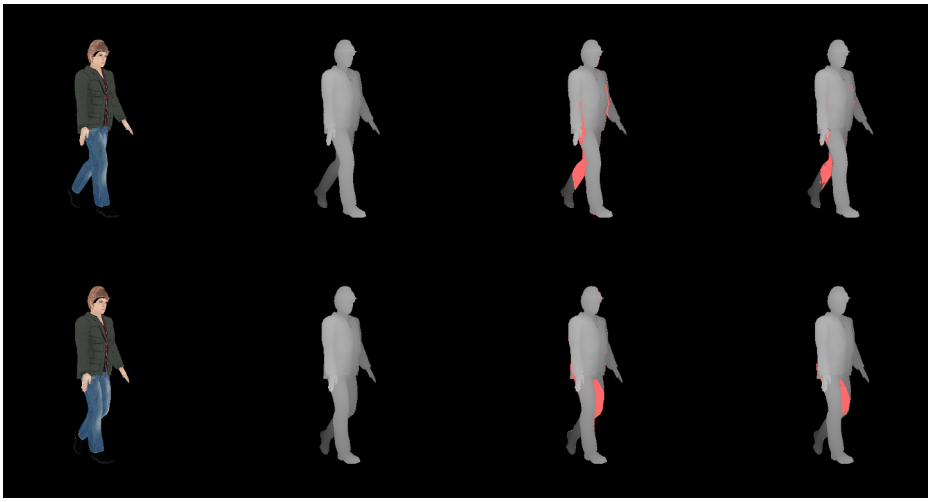

Figure 6. Occlusion Detection Example. From left to right we plot the RGB image, depth image, ground truth occlusions, and estimated occlusions for input frames 1 (top) and 2 (bottom). Occlusions are visualized as red regions superimposed on the whole depth image for reference. The occlusions in each row indicate pixels corresponding to parts of the body in that frame that are not visible in the other frame.

ter all, as we describe in Sec. 2.4, we are able to compute ground-truth occlusion maps from the ground-truth flow. However, flow estimates from the FlowNet are not as accurate as the ground truth. Thus, it is difficult for the OcclusionNet to estimate the occlusion map using only those estimates. On the other hand, incorporating the features—alone (B) or with the flow (D)—seems to improve the total performance (CD), even if flow estimation ($\widehat{OF}$) and feature-based correspondences do not improve or worsen.

Eliminating the sampling network improved the flow estimation results, as the U-net is now generating features specifically as inputs to flow estimation and occlusion detection. As expected, these features perform poorly for correspondence estimation. Even though the flow estimation is improved, the total performance becomes worse.

To evaluate the performance of occlusion detection, we show in Fig. 5 the receiver operating characteristic (ROC) curves, plotting the rate of occlusion detection (true positive rate) against false positive rate, as we vary a detection threshold in the output of the OcclusionNet.

The left plot compares the performance of combining flow estimates and features at the input of the OcclusionNet (table row D, blue curve in the figure) vs. using only the features (row B, green curve). We see significant performance improvement when flow is incorporated, which in a practical system should be weighed against the lower per-

formance in correspondence estimation, shown in Table 2.

The plot on the right illustrates in red the results of the ablation in row E of the table, in which we remove the sampling part of the network. The effect in occlusion detection is imperceptible, compared to including this component in the network. Given the benefits in correspondence estimation, there is no reason to remove it.

A qualitative evaluation of our occlusion detection is shown in Fig. 6. The top and bottom rows show the first and second input frames, respectively. From left to right we plot the RGB image, the depth image, the depth image with the ground truth occlusions highlighted in red, and the depth image with the estimated occlusions highlighted in red. The occlusions in each row indicate for this frame the pixels corresponding to parts of the body that are not visible in the other frame.

As evident in the picture, the detection is fairly high quality. The main issues we observed were around the edges of large occlusions. This is especially an issue when there is little motion and the occlusions are very few and not solid blocks. In the figure this is evident around the arms and the hands of the person. In contrast, the estimates around the legs are quite accurate.

3.3. Computational Considerations

Training the network for 120 epochs takes approx. three days on an RTX GPUs compute cluster with a batch size of 4 image pairs. Forward passes through the network for inference are fast. Although Eq. 6 has to be solved for all pixels in $\mathcal{P}_1$, this can easily be performed in parallel on a GPU, and this makes real-time inference possible.

4. Conclusion

We presented a scalable multi-modal network architecture for learning dense correspondences and detecting occluded pixels between subsequent frames of walking people. Our experiments show that combining contrastive loss with optical flow loss and using multiple modalities gives the best result. In particular, our flow inference approach uses the smoother optical flow estimate to regularize feature matching across frames and impose local smoothness constraints. This allows better filtering of spurious feature matching and improved performance. The estimated features and flow also facilitate occlusion estimation. Our architecture can be easily extended to additional modalities, such as infrared images, to further improve performance. Of course, it is important to validate our approach on real data, which have degraded optical flow ground truth.

References

[1] Optical flow accuracy and interpolation evaluation. https://vision.middlebury.edu/flow/eval/. 1

[2] B. Allain, J.-S. Franco, and E. Boyer. An Efficient Volumetric Framework for Shape Tracking. In *CVPR 2015 - IEEE International Conference on Computer Vision and Pattern Recognition*, pages 268–276, Boston, United States, June 2015. IEEE. 1

[3] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Int. Conf. Comput. Vis.*, pages 2758–2766, 2015. 1, 3

[4] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard. Multimodal deep learning for robust rgb-d object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 681–687, 2015. 2

[5] R. A. Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018. 1

[6] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. *IEEE Conf. Comput. Vis. Pattern Recog.*, abs/1612.01925, 2016. 1, 3, 6

[7] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018. 3

[8] T. Pfister, J. Charles, and A. Zisserman. Flowing convnets for human pose estimation in videos. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1913–1921, 2015. 2

[9] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[10] A. Shahroudy, T.-T. Ng, Y. Gong, and G. Wang. Deep multimodal feature analysis for action recognition in rgb+d videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:1045–1058, 2018. 2

[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In R. Cipolla, S. Battiato, and G. M. Farinella, editors, *Machine Learning for Computer Vision*, pages 119–135. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. 1

[12] R. Urtasun, D. J. Fleet, and P. Fua. Temporal motion models for monocular and multiview 3d human body tracking. *Computer Vision and Image Understanding*, pages 157–177, 2006. 1

[13] L. Wei, Q. Huang, D. Ceylan, E. Vouga, and H. Li. Dense human body correspondences using convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1544–1553, 2016. 2, 6

[14] W. Zeng, W. Ouyang, P. Luo, W. Liu, and X. Wang. 3d human mesh regression with dense correspondence. *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1