Improvement of Optical Flow Estimates by Visuomotor Anticipation

Proc. SOAVE2000 - SelbstOrganisation von Adaptivem VErhalten, Ilmenau, Fortschritt-Berichte VDI, Reihe 10: Vol. 643, pp. 14-21, 2000

Volker Stephan, Torsten Winkler

Department of Neuroinformatics, Ilmenau Technical University
98684 Ilmenau, Germany, P.O.B. 100565
volker.stephan@informatik.tu-ilmenau.de

Abstract

In this paper, we present a biologically inspired methodology to improve noisy estimates of optical flow by means of an internally generated expectation of the sensory consequenes of the system's own actions. Thereto we utilize a hybrid neural architecture to predict optical flow fields as consequences of the systems actions. A neural field-based approach is used to fuse sensory bottom-up and predicted top-down expectations. All subsystems use confidence estimates to reduce disturbances caused by noise. The facilities of this anticipative preprocessing can be demonstrated by means of an optical flow field based local navigation behavior of the miniature robot KHEPERA. Our anticipative preprocessing enables the robot to bridge gaps of sensory dropouts and, in consequence, to avoid collisions even with very noisy sensory information.

Keywords: optical flow prediction, expectation, sensor fusion, sensorimotor anticipation

1 Introduction

Traditional approaches to visual perception are based on the 'information processing paradigm' [Marr, 1982], which can be characterized by a strict separation between sensory perception and generation of behavior (see [Pfeifer and Scheier, 1994, Moeller and Gross, 1994] for a review). In recent years, the appreciation of visual perception as a generative sensorimotor process gained increasing acceptance [Cliff, 1990, Arbib et al., 1998]. The generative aspect of perception has been emphasized especially by [Kosslyn et al., 1993, Kosslyn and Sussman, 1995, Kosslyn, 1996] who supposed that internal simulation and mental imagery may play an integral role in perception, helping not only to recognize objects but also to anticipate the consequences of events. If this holds true at different levels of complexity and for different modalities, then, there must exist structures that are capable of predicting the sensory consequences of actions. Such sensory predictors seem to be multi-functional systems, since they can be used to a) enhance the incoming bottom-up sensory information by a top-down expectation generated previously b) direct selective attention to those environmental subregions which caused a mismatch of top-down expectation and bottom-up sensory information and c) internally simulate the consequences of action sequences in order to find and execute those actions, that entail positive outcomes for the system [Gross et al., 1999].

In this paper, we present a hybrid network architecture to predict optical flow fields and demonstrate its functionality in a robot-navigation task. This is done by means of a fusion of sensory bottom-up and expectation-based top-down information. This is a kind of anticipative preprocessing embedded in a cognitive processing cycle of hypotheses generation and verification [Kosslyn and Sussman, 1995, Kosslyn, 1996].

2 Experimental framework

For our experiments, we use the real robot platform KHEPERA, a miniature robot equipped with an omni-directional color-camera (see Figure 1, left) to demonstrate the improvement on the incoming sensory information stream by fusion with a top-down expectation. We believe, that the behavior of an autonomous system operating only on this information is a very good indicator of the performance of the system's 'perception' of its environment.
The system's goal is a collision-free local navigation based only on visual information, in this case, the optical flow field. We use optical flow, because it is largely independent of specific visual details of the objects in the scene and yields implicit information about spatial distances to objects.

Fig. 1: *Used robot platform* KHEPERA *equipped with an omni-directional camera (left). Top-view of the environment with the* KHEPERA *(right).*

In the preprocessing of the original omni-camera-images we perform a polar transformation (see Figure 2 left) to the deskewed form depicted in Figure 2 (right). These transformed images are used directly to estimate the optical flow fields, because an action of the robot with a rotational part yields a rotation of the omni-camera-image but only a shift in x-direction of the polar transformed image. This is very advantageous, since the applied correlation based optical flow estimation [Barron et al., 1994] needs not cope with rotated correlation areas, which would be very time consuming.

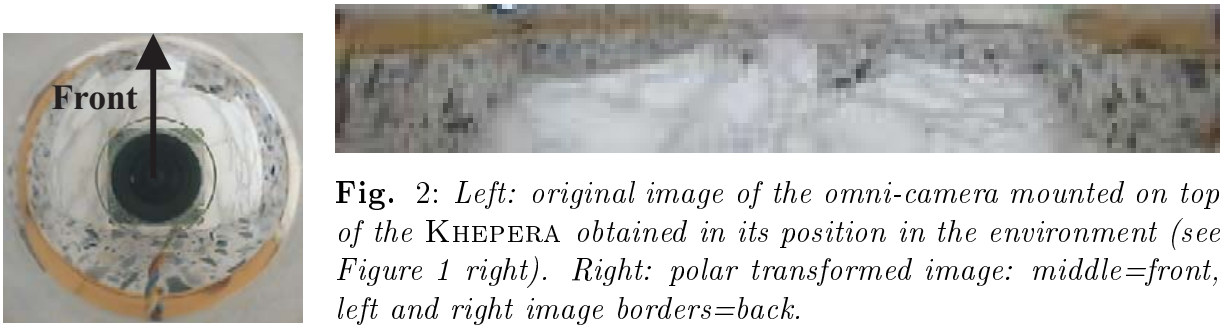

Fig. 2: *Left: original image of the omni-camera mounted on top of the* KHEPERA *obtained in its position in the environment (see Figure 1 right). Right: polar transformed image: middle=front, left and right image borders=back.*

3 Architecture

As introduced in section 1, we use a hybrid architecture to predict the optical flow fields as a result of the previous optical flow field and the real or hypothetical action to be executed. To demonstrate the functionality in a KHEPERA-scenario, we fuse the sensory bottom-up estimate and the top-down expectation in order to reduce the noise and gain robustness against sensory dropouts (see Figure 3).
A central aspect of our anticipative processing in the bottom-up/top-down cycle is the usage of flow vector specific confidence estimates organized topographically corresponding to the flow field.

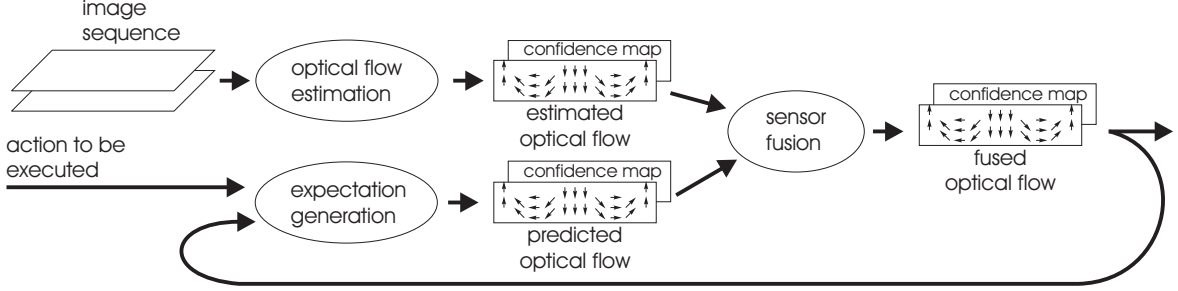

Fig. 3: *Hybrid architecture to fuse the sensory bottom-up and the top-down expectation.*

These confidence-values of each flow vector are based on correlation-based optical flow estimation [Barron et al., 1994] by evaluating the shape of the correlation function. Sharp and unique minima cause high confidence values, whereas flat or ambiguous correlation functions result in low ones.

3.1 Expectation generation

Sensorymotor prediction is of central importance for our approach. In previous approaches [Gross et al., 1999], we used standard neural networks, such as multilayer-perceptrons or a mixture of experts consisting of several action-specific perceptrons. In some cases, these networks had prediction problems, especially if unknown configurations of obstacles were presented. In our present view, the key problem of the used neural networks was the prohibitively high dimensionality of the sensory input, the whole optical flow field. However, a single optical flow vector to be predicted, depends only on a very small part of the current flow field, but never on the whole field. Hence, a network with completely connected layers first has to find the respective 'source-region' and thereafter to learn to predict the corresponding flow vector for each position of the flow field.
Hence, we developed an alternative approach to overcome this problem. It uses the property inherent to optical flow to represent the movements of objects onto the camera-plane. Consequently, the inverse optical flow itself points to that part of the image, where movement must be predicted (see Figure 4).

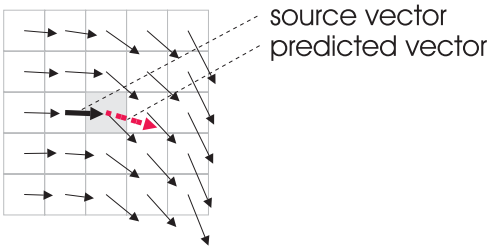

Fig. 4: *The flow vector to be predicted (dashed vector) depends only on those flow vector (bold), that point closest to the position for which the flow vector is to be predicted. That is, because this vector describes the velocity of the corresponding objects in the scene.*

Equation 1 shows the computation rule for a source vector $\underline{f}^S_{pq}(t)$ at position (p, q) within the optical flow field. This is a superposition of all vectors $\underline{f}^M_{kl}(t)$ depending on the superposition weight r_{pqkl}. r_{pqkl} reflects, how exactly these optical flow vectors point to the current position $(p, q)^T$ and is normalized by the sum of all superposition weights (see equation 2). The search for this source-vector among all flow vectors $\underline{f}^M_{kl}(t)$ can be reduced to a small region around the current position (k, l), where the size of this search-window corresponds to the size of the search-window of the optical flow estimate determined by n. Thereafter, the current predicted vector $\underline{f}^P_{pq}(t+1)$ results from the sum of the source-vector $\underline{f}^S_{pq}(t)$ scaled in x and y-direction by

the weights $w^S_{xpq}(t), w^S_{ypq}(t)$ and its previous value, $\underline{f}^M_{pq}(t)$ scaled by $w^O_{xpq}(t), w^O_{ypq}(t)$ respectively (equation 3). The confidences $c^P_{pq}(t+1)$ depend on the confidence $c^S_{pq}(t)$ of the source vector and on the confidence of the prediction itself $w^c_{pq}(t)$ (equation 5).

$$\underline{f}^S_{pq}(t) = \sum_{k=p-n}^{p+n} \sum_{l=q-n}^{q+n} \underline{f}^M_{kl}(t) \cdot r_{pqkl} \tag{1}$$

$$r_{pqkl} = \frac{\sum_{u=p-n}^{p+n} \sum_{v=q-n}^{q+n} \left\| \begin{pmatrix} u \\ v \end{pmatrix} + \underline{f}^M_{uv} - \begin{pmatrix} p \\ q \end{pmatrix} \right\|}{\left\| \begin{pmatrix} k \\ l \end{pmatrix} + \underline{f}^M_{kl} - \begin{pmatrix} p \\ q \end{pmatrix} \right\|} \tag{2}$$

$$\underline{f}^P_{pq}(t+1) = \begin{bmatrix} w^S_{xpq}(t) & 0 \\ 0 & w^S_{ypq}(t) \end{bmatrix} \underline{f}^S_{pq}(t) + \begin{bmatrix} w^O_{xpq}(t) & 0 \\ 0 & w^O_{ypq}(t) \end{bmatrix} \underline{f}^M_{pq}(t) \tag{3}$$

$$c^S_{pq}(t) = \sum_{k=p-n}^{p+n} \sum_{l=q-n}^{q+n} c_{kl}(t) \cdot r_{pqkl} \tag{4}$$

$$c^P_{pq}(t+1) = w^c_{pq}(t) \cdot c^S_{pq}(t) \tag{5}$$

The update of the weights depends on the difference between the predicted vector $\underline{f}^P_{pq}(t+1)$ and the one actually experienced in the next time step $\underline{f}^M_{pq}(t+1)$ (equation 6). The confidence-weights $w^c_{pq}(t)$ are updated according equation 7.

$$\begin{aligned} \Delta w^S_{xpq}(t) &= \eta \cdot \left(f_{xpq}(t) - f^P_{xpq}(t) \right) \cdot f^S_{xpq}(t-1) \\ \Delta w^S_{ypq}(t) &= \eta \cdot \left(f_{ypq}(t) - f^P_{ypq}(t) \right) \cdot f^S_{ypq}(t-1) \\ \Delta w^O_{xpq}(t) &= \eta \cdot \left(f_{xpq}(t) - f^P_{xpq}(t) \right) \cdot f^M_{xpq}(t-1) \\ \Delta w^O_{ypq}(t) &= \eta \cdot \left(f_{ypq}(t) - f^P_{ypq}(t) \right) \cdot f^M_{ypq}(t-1) \end{aligned} \tag{6}$$

$$\Delta w^c_{pq}(t) = \eta \cdot e^{-\frac{\left\| \underline{f}^M_{pq}(t+1) - \underline{f}^P_{pq}(t+1) \right\|}{2}} - w^c_{pq}(t) \tag{7}$$

The prediction of a flow field containing 18×5 vectors with the subsequent weight update is very fast (only $35ms$ on Pentium 200MHz).

3.2 Fusion

With regard to Figure 3, in this section we present the fusion between bottom-up and top-down information (see Figure 5). Each vector of the whole field is represented by a small 2-dimensional neural field, where the position within the neural field codes the x- and y-components of the flow-vector as a blob, and the activation of the blobs in the neural field is a measure for the corresponding confidence of this local flow vector. Due to this 2-dimensional representation, it is possible to hold many alternative hypotheses (blobs) for each flow vector. Consequently, both the sensory bottom-up and the top-down expectation can add their hypotheses about the real optical flow vector into the corresponding neural field, whereby similar hypotheses result in a superposition of the blobs at the same position.

For reasons of simulation resources, we split the 2-dimensional neural field into 2 one-dimensional neural vectors representing the x- and y-direction of the flow vector separately (equations 8, 9). Thus, the time to fuse two flow fields containing 18×5 vectors each could be reduced to $275ms$ (Pentium 200MHz). Equation 8 shows, that the new state in the fusion-map $\underline{z}^x_{pq}(t)$ is computed by the superposition of the discounted previous state $\underline{z}^x_{pq}(t-1)$ with $\alpha \in (0 \ldots 1)$, the

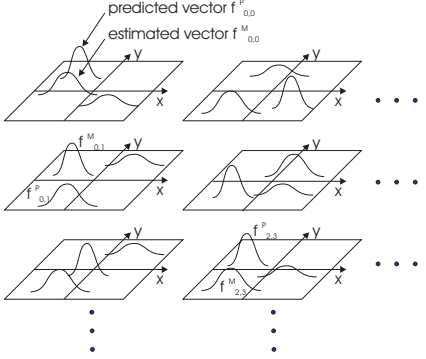

Fig. 5: *Each vector of the optical flow field is represented by a 2-dimensional neural field, where the position within the neural field codes possible flow-vectors as blobs, and the activation of the blob is a measure for the corresponding confidence of that respective optical flow vector.*

sensory bottom-up vector $\underline{g}(f^E_{xpq}(t))$ and the top-down expectation $\underline{g}(f^P_{xpq}(t))$ in form of 1 dimensional gaussian blobs (equation 10) weighted by their confidences $c(.)$ These 1 dimensional blobs realize a topological coding of the sharp x and y coordinates of the corresponding flow vectors and allow to represent multimodal hypotheses. The merger-output $\underline{f}^M_{pq}(t)$ results from the hypothesis with the highest confidence (equation 11).

$$\underline{z}^x_{pq}(t) = \alpha \underline{z}^x_{pq}(t-1) + \underline{g}(f^E_{xpq}(t)) \cdot c(f^E_{xpq}(t)) + \underline{g}(f^P_{xpq}(t)) \cdot c(f^P_{xpq}(t)) \quad (8)$$

$$\underline{z}^y_{pq}(t) = \alpha \underline{z}^y_{pq}(t-1) + \underline{g}(f^E_{ypq}(t)) \cdot c(f^E_{ypq}(t)) + \underline{g}(f^P_{ypq}(t)) \cdot c(f^P_{ypq}(t)) \quad (9)$$

$$g_k(u) = e^{-\frac{(u-k)^2}{2\sigma^2}} \quad (10)$$

$$\underline{f}^M_{pq}(t) = \begin{pmatrix} argmax_x(\underline{z}^x_{pq}(t)) \\ argmax_y(\underline{z}^y_{pq}(t)) \end{pmatrix} \quad (11)$$

Hence, this algorithm selects those of all hypotheses, which support each other. This is reasonable, since similar information in both streams implies, that this information is reliable and trustworthy.

4 Results

To train the flow field predictor, we put the robot KHEPERA into an environment similar to that depicted in Figure 1 (right) and let it cruise around for about 3 minutes. During this training period, the KHEPERA experienced several optical flow field configurations with obstacles on the left, on the right, and, also in front of it. These data were presented to the predictor network several times in order to train it (see figure 6 left). Figure 6 shows the decreasing prediction error over the training period (left) and some of the learned weight matrices (middle and right).

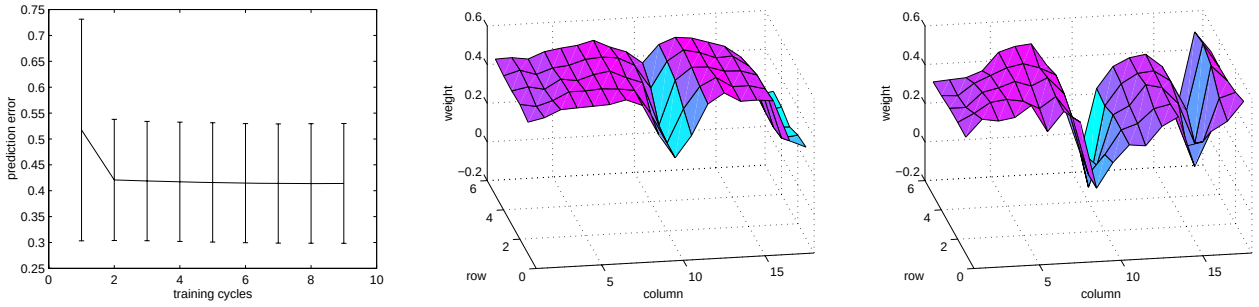

Fig. 6: *Left: development of prediction error during 9 training cycles. As can be seen, the predictor network learns very fast. Learned arrays of weights of our sensory predictor* $\underline{w}^S_x$ *for driving straight forward (middle) and for turning to the left (right). For detailed explanations see text.*

In the case of moving forward, the predictor amplifies optical flow vectors close to the central focus of expansion (FOE) and shortens lateral and vectors behind the robot (the weights in columns $6, 7$ *and* $12, 13$, are larger than those in columns $1 \ldots 4$ *and* $15 \ldots 18$). That is very

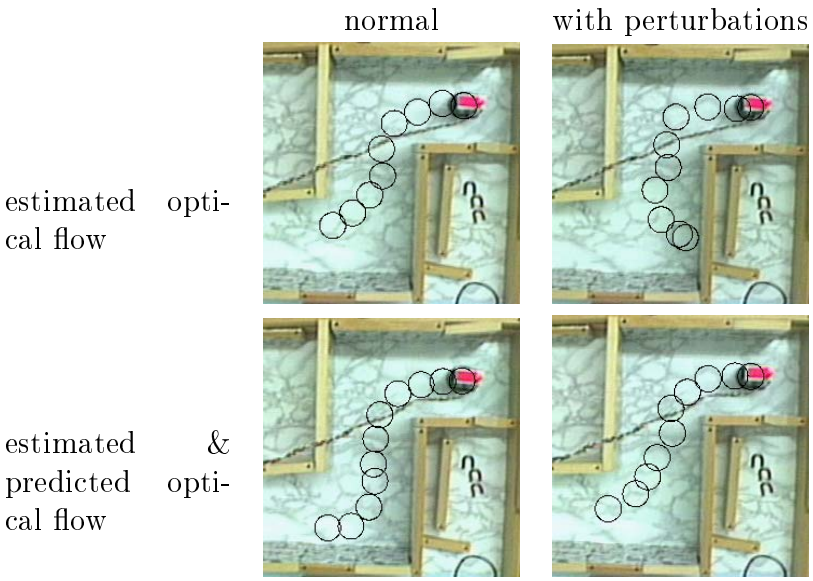

Fig. 7: *Navigation based on the estimated optical flow applying the well known balancing approach [Duchon et al., 1995] starting at the upper right corner and moving to the opposite one of the images. As can be seen, both the navigation on the pure estimated optical flow (top left) and on the expectation driven preprocessed optical flow (bottom left) allow a collision-free locomotion of the robot* KHEPERA *through the environment. In contrast, a significant disturbance of the optical flow estimate by fluctuating ambient light causes a collision at the end of the plotted trace, where no anticipative preprocessing is applied (top right). The anticipative preprocessing overcomes the problems and allows a collision-free locomotion (bottom right)*

plausible, since the projection of static objects onto the moving omnidirectional camera follows this principle. Central columns $(9, 10)$ around the FOE have very small weights, because these flow vectors are close to zero in most cases. In the case of the left turn (Figure 6 right), the typical shape of the drive-forward-weight-matrix is compressed in the right hemisphere and expanded in the left due to the self-rotation of the robot.

To demonstrate the facilities of the presented anticipatory preprocessing, we placed the robot in unknown environments to navigate through a narrow passage without collision. For this benchmark , we used the balancing approach [Duchon et al., 1995], which tries to equalize the optical flow in both hemispheres of the robot, which results in a collision-free locomotion in the middle of such an hallway. Figure 7 (left column) shows a top view of this scenario with collision-free traces of our robot. If a perturbation is applied in this experimental situation, the usage of pure estimated optical flow fields fails, because the very noisy sensory input entails no information about close obstacles. In contrast, our anticipatory preprocessing allows the system to bridge the time gap of sensory dropouts with the generated expectation and is therefore able to extract relevant information in order to avoid the arising obstacles.

At this point, we have to ask the question: would a pure feedback without any sensory prediction (see Figure 3) result in the same behavior? In this case, the fusion of the noisy and very unreliable estimated optical flow and the relatively reliable expectation would return the last fusion output. Thus, such an architecture is nothing but a low-pass filter over time. The advantage of our anticipative preprocessing in contrast to this approach is illustrated exemplarily in Figure 8. In this case, the robot stood in front of an obstacle at a distance of about 15cm

and drove straight forward. As can be seen, the system with anticipative preprocessing is able to recursively predict the enlarging central flow vectors representing the close obstacle, whereas the system without sensory prediction once again can only store the last sensory situation, which becomes more and more obsolete over time.

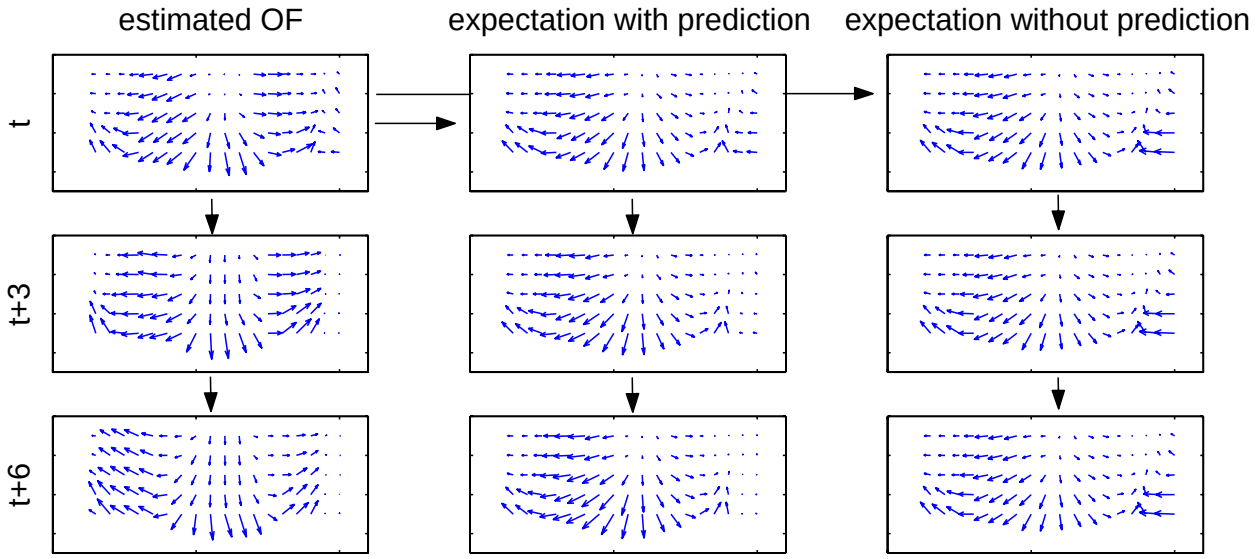

Fig. 8: *Sequences of optical flow fields for driving straigth forward of 9 subsequent steps of movement, where only every third is plotted to show the differences more clearly. Left column: sequence of estimated flow fields; middle: internally simulated sequence of flow fields starting from the estimated flow field in the first row. Each predicted flow field is the result of a confidence controlled superposition of the top-down prediction and the previous fused optical flow map. Right: same as in the middle, except that in this case, the top-down expectation is the previous fused optical flow field instead of a predicted one. As can be seen in the estimated flow maps (left column), a frontal obstacle causes* **increasing optical flow vectors in the central part** *of the vector map. The usage of the sensory predictor as expectation generator allows an adaptation to the oncoming object. In contrast, the approach without anticipatory preprocessing (right) does not reflect any changes in the optical flow. Hence, the sensory predictor is an essential part of our anticipative preprocessing.*

5 Conclusions and Outlook

In this paper, we presented a hybrid neural architecture to predict optical flow fields as consequences of actions. Further, we introduced a neural field-based method to fuse sensory bottom-up estimates and top-down expectations. All proposed subsystems extensively use confidence measurements in order to prevent disturbance by noise.

Even though the presented architecture shows various similarities to the well known Kalman filter approach [Welch and Bishop, 1997], there exist some essential differences about the fusion of a measurement and the corresponding prediction. In the case of the Kalman filter the fusion is a weighted superposition of the two datastreams, whereas our architecture selects the most reliable hypothesis from the neural field. Thus, interpolation only occurs between sufficiently similar hypotheses, whereas very different hypotheses are not merged as they would be with the Kalman filter. Instead, the most trustworthy supposition is preferred. In consequence,

our architecture is more robust against outliers. A detailed comparison with the Kalman-filter approach is subject of future work.
The facilities of the anticipative preprocessing could be demonstrated by means of a local navigation behavior of the real robot platform KHEPERA. The presented sensory prediction can be very useful for various tasks, such as the dynamic control of visual attention to regions, where a mismatch of expectation and sensation occurred, or the internal simulation and evaluation of longer action sequences in order to find an optimal action sequence according to the current system state [Gross et al., 1999].
Future work will address the improvement of sensory prediction. The current network causes problems with large steering angles and cannot cope with different speeds of the robot. Moreover, problems emerging from object occlusions have to be solved, to allow the robust prediction of longer sequences in order to apply the predictor network to our model for anticipation based on sensory imagery.

References

[Arbib et al., 1998] Arbib, M., Erdi, P., and Szentagothai, J. (1998). *Neural Organization: Structure, Function and Dynamics.* MIT Press.

[Barron et al., 1994] Barron, J., Fleet, D., and Beauchemin, S. (1994). Performance of Optical Flow Techniques. *International Journal of Computer Vision, 12:1*, pages 43–77.

[Cliff, 1990] Cliff, D. (1990). *Computational Neurothology: A Provisional Manifesto.* University of Sussex, School of Cognitive and Computing Sciences.

[Duchon et al., 1995] Duchon, A., Warren, W., and Kaelbing, L. (1995). Ecological Robotics: Controlling Behavior with Optical Flow. *Proceedings of the 17th Annual Cognitive Science Conference. J.D. Moore and J.F. Lehman (eds.) Lawrence Erlbaum Associates.*, pages 164–169.

[Gross et al., 1999] Gross, H.-M., Heinze, A., Seiler, T., and Stephan, V. (1999). Generative Character of Perception: A Neural Architecture for Sensorimotor Anticipation. *Neural Networks*, 12:1101–1129.

[Kosslyn, 1996] Kosslyn, S. (1996). *Image and Brain: The Resolution of the Imagery Debate.* MIT Press.

[Kosslyn et al., 1993] Kosslyn, S., Alpert, N., and Thompson, W. (1993). Visual mental imagery activates topographically organized visual cortex: PET investigations. *Journal of Cognitive Neuroscience*, 5(3):263–87.

[Kosslyn and Sussman, 1995] Kosslyn, S. and Sussman, A. (1995). Roles of Imagery in Perception: Or, There is No Such Thing as Immaculate Perception. In Gazzangia, M., editor, *The Cognitive Neuroscience*, pages 1035–1042. MIT Press.

[Marr, 1982] Marr, D. (1982). *Vision.* San Francisco: Freeman.

[Moeller and Gross, 1994] Moeller, R. and Gross, H.-M. (1994). Perception through Anticipation. In *Proc. PerAc'94 - From Perception to Action*, pages 408–411. Los Alamitos: IEEE Computer Society Press.

[Pfeifer and Scheier, 1994] Pfeifer, R. and Scheier, C. (1994). From Perception to Action: The Right Direction? In *Proc. PerAc'94*, pages 1–11. IEEE Computer Society Press.

[Welch and Bishop, 1997] Welch, G. and Bishop, G. (1997). An Introduction to the Kalman Filter. Technical report, Department of Computer Science, University of North Carolina.