Self-Organizing, Adaptive Data Fusion for 3d Object Tracking

Olaf Kähler, FSU, Dept. Mathematics and Computer Science, Chair for Computer Vision, 07743 Jena, Germany
Joachim Denzler, FSU, Dept. Mathematics and Computer Science, Chair for Computer Vision, 07743 Jena, Germany

Abstract

Data fusion concepts are a necessary basis for utilizing complex networks of sensors. A key feature for a robust data fusion system is adaptivity, both to be fault-tolerant and to run in a self-organizing manner. In this contribution a general framework for adaptive data fusion is established with object tracking as an application. The fusion algorithm of Democratic Integration is presented as one possible robust approach to the fusion task. As an alternative the STAPLE algorithm will be shown, which was previously only used for late classifier fusion. Extensions to apply the STAPLE algorithm for the fusion of probabilities will be introduced. Finally both algorithms will be evaluated on complex, realistic scenes to show their capabilities of self-organization and fault-tolerance.

1 Introduction

One of the key concepts for handling large collections of autonomous systems with sensors is data fusion. The information gathered by the individual sensors has to be combined in an intelligent manner. This combination necessarily has to be fault-tolerant, context-aware and self-organizing. We present a general framework for an adaptive data fusion and two algorithms to actually perform the fusion within this framework. The presented methods were implemented with the application of 3d object tracking in mind but can easily be applied to different problems.

According to the classification in [1] the framework used as backbone of our system is capable of handling *competitive* and *cooperative* fusion. This means in case of contradicting data there will be a self-organizing competition for the best result. Also the sensors will cooperate to reconstruct 3d position estimates, which could not be achieved by any of the sensors alone.

Other categories to differentiate data fusion systems exist in the literature [2], e.g. into *early* and *late* fusion. In early fusion the data is first combined and then evaluated as a whole, while in late fusion the data is evaluated independently and then the decisions are combined. We use an intermediate solution called probabilistic fusion. The data is evaluated independently as in late fusion, which makes the approach flexible for combining different classes of data. No hard decisions are enforced however but the individual results are combined probabilistically which allows uncertainty of the decisions to be preserved.

Our framework relies on previous works presented in [3, 4]. In 2d images the state space representing where the tracked object resides will be completely represented by a map of saliencies or probabilities. In case of 3d world coordinates a particle filter based approximation of the state space and thus a sampling of the probability distribution was proposed. The data fusion concept of Democratic Integration [5] was used in these earlier works. This concept will be reviewed here as example for a complex, self-organizing fusion system.

As the algorithm of Democratic Integration is not inspired by a mathematical formalism but by plausible biological observations, the need for a different method arises. The STAPLE Algorithm (Simultaneous Truth and Performance Level Estimation) proposed in [6, 7] will be introduced and reformulated in the context of our probabilistic framework. As mathematical background for this method only the well known Bayes formula is used.

Both algorithms have means to adapt in a self-organizing manner to changing environment conditions. The strengths and weaknesses of their adaptation will be evaluated theoretically first and then by experimental comparison for the object tracking application.

2 Framework for Data Fusion

First the probabilistic framework for the data fusion will briefly be explained. These concepts can be transferred to different applications apart from the visual tracking task used in this work. We show that data acquired from different state spaces can efficiently be fused within one combined state space. In our work this is used to fuse multiple 2d views of a scene into a 3d hypothesis of the position of a tracked object.

Elements $\boldsymbol{x}$ in the respective 2d or 3d state spaces can either belong to the tracked object, in which case they are elements of a class Ω_1, or they are part of the background and elements of class Ω_0. In the simple case of 2d cues each state $\boldsymbol{x}$ represents one pixel in the images. As an input for the system we have sensor data $\boldsymbol{s_j}$ from the sensors $j = 1 \dots J$. To process this raw data and retrieve information in the state space we further have cues p_k detecting salient regions in the images. The 2d cues then use methods $\mathcal{M}_k$ with parameters $\boldsymbol{r}_k$ to assign a probability of belonging to Ω_1, i.e. the tracked object, to each pixel or state:

$$p_k(\boldsymbol{x} \in \Omega_1) = \mathcal{M}_k(\boldsymbol{x}, \boldsymbol{s}_j, \boldsymbol{r}_k) \tag{1}$$

Typical examples for basic input cues used in this and other works [5] are motion detection by pixel based difference

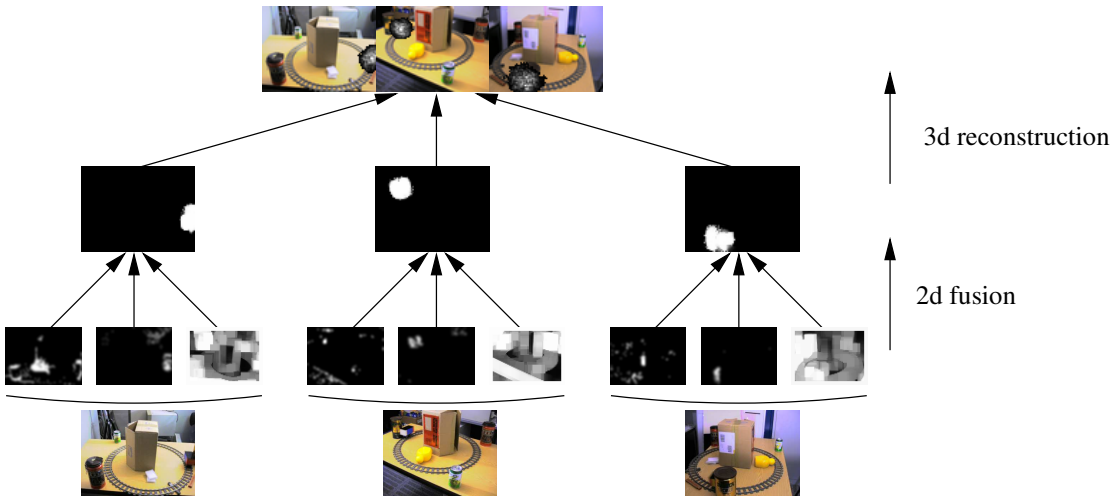

Figure 1 Example for hierarchical data fusion using three sensors with three cues each. The cues give estimations on a tracked objects position in the respective state spaces (image planes). Difference image, color tracking and contrast cues are used in the example above.

images, detecting an area with a specific color (color tracking), computing the correlation of a template and the pixels in an image (template matching) and finally finding regions with a typical contrast.
In a data fusion step several of these cues $p_k, k = 0, \ldots, K$ can now be combined to a new cue p_c. For the fusion step itself the arrangement of the states or pixels is irrelevant, i.e. the fusion of the ratings for one pixel is independent of the fusion of neighboring pixels. Therefore the combination is expressed in the equation

$$p_c(\boldsymbol{x} \in \Omega_1) = \mathcal{C}(p_1(\boldsymbol{x} \in \Omega_1), \ldots, p_K(\boldsymbol{x} \in \Omega_1), \boldsymbol{r}_c) \quad (2)$$

where the combination function $\mathcal{C}$ and its parameters $\boldsymbol{r}_c$ will be detailed in the following sections. Note a fusion step can itself be used as input cue for another fusion step, which was applied in [4] to build a hierarchical fusion system. In this work we also typically use this hierarchical structure as depicted in figure 1.
An open issue so far is the combination of different state spaces, i.e. in our case the combination of several 2d views to reconstruct 3d positions. In [3] an approach using particle filters was introduced. The state space of the 3d object position is represented as a set of particles and the condensation algorithm [8] is applied. To calculate weights or ratings for the particles each of them is projected into the respective image planes of the fused inputs using projections $\boldsymbol{\pi}_k$. The ratings thus gained from several data sources can then be combined as before. The projections $\boldsymbol{\pi}_k$ can be determined using a camera calibration step [9]. Overall such a combination of different state spaces can be described by the formula

$$\begin{aligned} &p_c(\boldsymbol{x} \in \Omega_1) \\ &\quad = \mathcal{C}(p_1(\boldsymbol{\pi}_1(\boldsymbol{x}) \in \Omega_1), \ldots, p_K(\boldsymbol{\pi}_K(\boldsymbol{x}) \in \Omega_1), \boldsymbol{r}_c) \end{aligned} \quad (3)$$

with the same combination functions $\mathcal{C}$ and parameters $\boldsymbol{r}_c$ as in the pure 2d case.

To achieve a self-organizing data fusion framework we finally add an adaptation step to the system. Thereby the globally best known result is fed back into the individual fusion steps and cues to adapt their internal parameters. Examples for these parameters are the tracked templates and colors or in case of a data fusion the reliabilities of the fused input cues. Using the global result this mechanism achieves a weak coupling between the cues. The results of high valued sensors and cues can influence and possibly correct the less valued cues. Also user defined controls could be injected into the system this way by giving a different adaptation goal.
Having a global estimate of the 3d position the same projections $\boldsymbol{\pi}_k$ as before can be used to define the adaptation goal on the individual 2d cues. The particles representing the probability distribution in the higher dimensional state space have to be projected into a 2d distribution, which again must be handled by the individual adaptation mechanisms of the cues.
The overall framework is schematically shown in figure 2. Starting with the raw data some input cues first try to detect salient regions in the image. Several inputs are then combined and the crude initial estimations are refined by the combination step. From this combined result a state representing the position of the tracked object can be selected. Alternatively the result can also be used in a next hierarchy step, as in figure 1. Finally in an adaptation step internal parameters of both data fusion and fused input cues can be adapted by a feedback of the global estimation.
Having presented the inputs and a general framework for the self-organizing sensor data fusion the specific combination functions $\mathcal{C}$ and their respective adaptation mechanisms yet have to be defined. The two algorithms of Democratic Integration and STAPLE-Fusion fit into this framework providing both a possibility for combining data and for adapting internal parameters to re-weight the individual

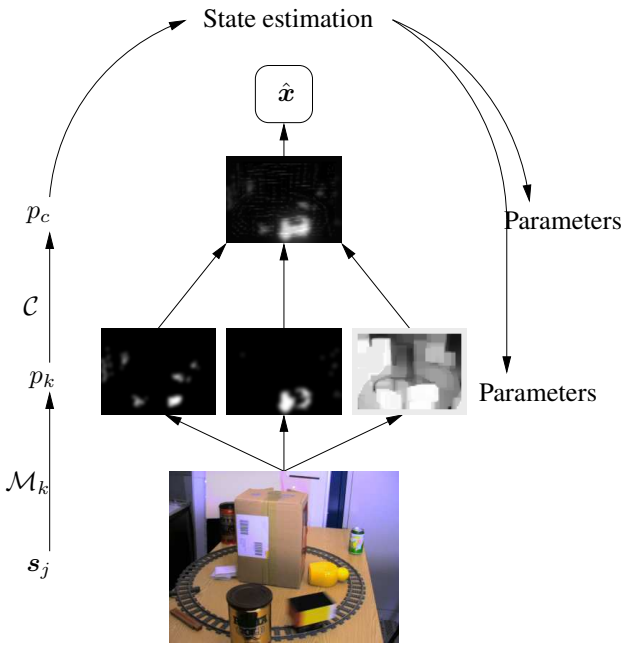

Figure 2 Overview of the data fusion framework for 2d images including an adaptation step.

contributions.

3 Democratic Integration

The idea of Democratic Integration has been introduced in [5] and was originally biologically inspired. The algorithm has since been studied in greater detail, e.g. in [3, 4]. In previous works a one-state-hypothesis was implicitly assumed. Using this assumption practically the tracked object can only be at exactly one position at a time. Each cue then computes a probability distribution over the whole state space of where the object could be. The tracked object can then be located with the maximum likelihood criterion. To see it even more practically this means each cue output is normalized to sum to 1 over the whole state space before further processing.

In Democratic Integration a weighted sum is used as combination function for the cues. Therefore to compute the combined probability p_c from the inputs $p_1, \ldots, p_K$ the following expression is evaluated:

$$p_c(\boldsymbol{x}) = \sum_k w_k p_k(\boldsymbol{x}) \tag{4}$$

Weighted voting mechanisms have been used previously in data or classifier fusion [2]. However in the Democratic Integration scheme another step is added to autonomously organize the weights in an intelligent way. An input is needed for this adaptation step, which is typically the fusion output p_c itself, as was mentioned in the description of the framework. Therefore a self-organizing system is achieved.

In the adaptation step first the quality q_k of each fused cue is calculated. A high quality is assigned to cues having a high agreement with the global result, while cues disagreeing with what is thought to be the optimal distribution receive a low quality. Various typical distance measures like sum of squared differences, cross-correlation or Kullback-Leibler divergence can be used and have been compared experimentally before [5, 4]. No clearly superior measure could be determined however.

Using the qualities and an adaptation rate τ the weights can be modified:

$$w_k := \tau q_k + (1 - \tau) w_k \tag{5}$$

Altogether a robust object tracking system can be based on this self-organizing algorithm. In the experiments of previous works a very good performance was shown also in complex scenes with the tracked object being temporarily occluded and more difficulties. However the approach leaves a somewhat heuristical impression and its good performance can hardly be explained from a mathematical point of view. Effort was therefore put into the second presented approach trying to handle data fusion in a more mathematically established formulation.

4 STAPLE-Fusion

As completely different algorithm the STAPLE-Fusion was introduced in [6] and since has been refined e.g. for efficient segmentation with more than two classes [7]. Yet it has only been used in the context of late classifier fusion i.e. with binary decisions, and not in a probabilistic fusion as we are aiming at in our work. The main drawback of late fusion is the missing ability to handle uncertain input information. Instead of probabilities hard decisions are enforced as inputs to these algorithms.

For sake of simplicity the self-organizing structure of the STAPLE-Fusion will first be explained with such binary decisions e_k, which can be computed from probabilities p_k by maximum likelihood estimation. In practice this corresponds to a binarization of p_k. After the basic idea is made clear with this simplified fusion task we will propose a way to get around the hard decisions.

4.1 Binary Decisions

The STAPLE-Fusion is based on the EM-Algorithm which is a classical approach for unsupervised learning. Two steps called *Expectation* and *Maximization* are iterated which in our case first combine the data and then refine the combination parameters using the results just calculated. For the dynamic image sequences we do not iterate these steps until convergence for each image but use the consecutive image in each next iteration step. Thus we reach a similar series of processing steps as with the Democratic Integration approach.

The combination step is motivated using the well known Bayes-Formula. Depending on the input decisions e_k we want to calculate the probability p_c of a state $\boldsymbol{x}$ to belong to the tracked object Ω_1. This can be transformed to several

a priori probabilities and the probability of the individual decisions knowing the real membership of the state $\boldsymbol{x}$:

$$p_c(\boldsymbol{x} \in \Omega_1) = P(\boldsymbol{x} \in \Omega_1 | e_1(\boldsymbol{x}), \ldots, e_K(\boldsymbol{x})) = \frac{P(\boldsymbol{x} \in \Omega_1) P(e_1(\boldsymbol{x}), \ldots, e_K(\boldsymbol{x}) | \boldsymbol{x} \in \Omega_1)}{\sum_{i=0,1} P(\boldsymbol{x} \in \Omega_i) P(e_1(\boldsymbol{x}), \ldots, e_K(\boldsymbol{x}) | \boldsymbol{x} \in \Omega_i)} \quad (6)$$

Assuming the stochastical independence of the decisions e_k equation 6 further simplifies to

$$p_c(\boldsymbol{x} \in \Omega_1) = \frac{P(\boldsymbol{x} \in \Omega_1) \prod_k P(e_k(\boldsymbol{x}) | \boldsymbol{x} \in \Omega_1)}{\sum_{i=0,1} P(\boldsymbol{x} \in \Omega_i) \prod_k P(e_k(\boldsymbol{x}) | \boldsymbol{x} \in \Omega_i)} \quad (7)$$

The missing variables to calculate p_c are therefore the a priori probabilities $P(\boldsymbol{x} \in \Omega_i)$ and the sensitivities and specificities $P(e_k(\boldsymbol{x}) = i' | \boldsymbol{x} \in \Omega_i)$ representing the cue reliabilities. The EM-like idea presented in [6] is to calculate approximates of these probabilities by counting their respective occurrences assuming p_c as a given optimal allocation $\boldsymbol{x} \in \Omega_i$. This means counting the individual decisions assuming known membership in the classes Ω_0 and Ω_1. An adaptation rate τ can be introduced as before to smoothen the reliabilities in the case of changing input data in each iteration step. With constant input data and no additional adaptation rate convergence was shown [6].

4.2 Non-binary Decisions

The use of sensitivities and specificities $P(e_k(\boldsymbol{x}) = i' | \boldsymbol{x} \in \Omega_i)$ requires a binary decision model. Within our framework we want to propagate uncertainties and therefore probabilities. A different decision error model has to be used to achieve this. As the inputs for our fusion are real valued in $[0; 1]$ the decision-probabilities $P(p_k(\boldsymbol{x}) | \boldsymbol{x} \in \Omega_i)$ have to be defined on the whole interval $p_k(\boldsymbol{x}) \in [0; 1]$.

The theoretically correct approach to approximate the curve on the whole interval is to use a Parzen estimation. This method is computationally very complex however and for real-time tracking applications different solutions have to be found.

Another idea is to approximate the decision-probabilities by histograms with D bins and linear interpolation between the center points of the intervals. It has shown however that these histograms can not be estimated robustly for the whole interval $[0; 1]$ as decisions near the extremes of this interval are extremely rare in a practical implementation. Using the simple and fast counting based adaptation step is not possible due to the lack of data in some of the intervals.

The currently best solution is to use the following expressions instead of the sensitivity and specificity of a given decision z:

$$\begin{aligned} &P(p_k(\boldsymbol{x}) < z | \boldsymbol{x} \in \Omega_1) \\ &P(p_k(\boldsymbol{x}) > z | \boldsymbol{x} \in \Omega_0) \end{aligned} \quad (8)$$

This also solves the problem that $P(p_k(\boldsymbol{x}) = z | \boldsymbol{x} \in \Omega_i) = 0$ as imposed by probability theory of continuous distributions.

Using half-open intervals however is only valid under the assumption that basically all fused cues are well-natured. This is meant in the sense that they tend to rather decide on high values $p_k(\boldsymbol{x} \in \Omega_1)$ if the state really belongs to Ω_1. For our decision-probabilities $P(p_k(\boldsymbol{x}) | \boldsymbol{x} \in \Omega_i)$ this means a monotonic increase.

In our implementation we again used an histogram based approximation of the newly defined sensitivities and specificities with different numbers of intervals D. Better solutions might be possible, good experimental results have already been found with this simple one however. This way the half-open decision-probabilities can be efficiently implemented which is a necessary precondition in all real-time object tracking tasks.

Altogether with both of the presented data fusion methods a robust object tracking system can be implemented within the defined probabilistic framework. Special emphasis was put on the adaptation steps of both algorithms. With the feedback loop a self-organizing system is established. The system as a whole can dynamically decide which cues to rely on. It is therefore fault-tolerant or in another sense self-healing, with the healing process resulting from the feedback of an appropriate adaptation goal. A more detailed comparison of the two algorithms and especially practical experiments to show applicability of the theory is given in the following section.

5 Comparison

Different approaches are used for a comparison of the two mentioned algorithms. First the behavior of the data fusion algorithms can be analyzed and predicted with some theoretical considerations. This will provide further insights into the mathematics of data fusion. Practical applicability can only be demonstrated with real image sequences however. Results for a complex object tracking task with ground truth data available will therefore be presented as a second section.

5.1 Theoretical comparison

The performance of the mentioned fusion algorithms can easily be theoretically analyzed for two naive but frequent situations. The first is the handling of a totally uninformed input cue or sensor, or more general the influence of uncertainty on the fusion result.

With the weighted sum in Democratic Integration uncertainty has a weakening influence on the global estimation. Consider the situation where three out of four equally weighted cues are absolutely sure to have the object in one state $\boldsymbol{x}$. The fourth cue however is totally unsure about whether the object is in this state. In the sum the fourth cue weakens the decision of the other cues, although not knowing anything about state $\boldsymbol{x}$. Considering the Bayes formula in the STAPLE fusion a cue deciding for both classes, object and background, with equal weight can be ignored altogether as it is canceled out in the fraction. This consideration can easily be extended as to say within the

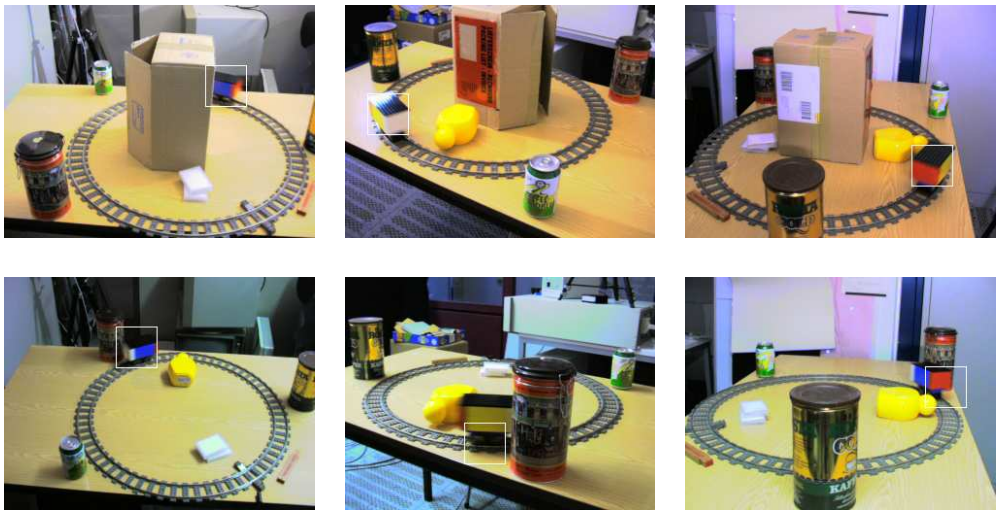

Figure 3 Examples from the test sequences used in the experiments, `seq6_light` on top and `seq8` below. Three cameras with different viewpoints observe a toy train moving through the scene. The estimated 3d position of the train is backprojected into the images and indicated by the rectangles.

STAPLE fusion uncertain input data can be discarded in contrast to weakening the weighted sum of Democratic Integration. The behavior of ignoring undecided cues seems more intuitive than allowing influence on the fusion result. When dealing with multiple inputs another frequent situation are outliers. If due to misbehavior a single cue assigns a completely different saliency to a state than all others, the influence of this contribution in the Democratic Integration approach is still bounded by the weight of the cue. In the multiplicative terms of the STAPLE fusion a single cue contributing a 0 could however introduce a veto such that all other cues are ignored and the combined result will be 0 as well.

Concluding the theoretical comparison we can expect a better ability to ignore uncertain cues or in other words more decisive results for the STAPLE fusion. However in cases where many outliers occur and the adaptation mechanism has not yet rearranged the weights of the fused cues, the Democratic Integration approach can be expected to run more stable without getting confused. In the following section we observe exactly this behavior in our practical experiments.

5.2 Experimental Comparison

A major consideration for the experimental setup was the ability to get ground truth data to be able to numerically compare the performance of different data fusion settings. Other than that a complex setup was chosen to show robustness and adaptivity also under extreme conditions.

Basically three cameras observe a scene with a toy train moving on a circular track. The camera positions as well as the circle defining the motion were first computed by calibration with manual interaction. Ground truth of the position of the tracked train in each frame of the sequences could thus be obtained. Calculating the differences of the estimated positions to the positions modeled by the circular movement allows an objective measure for the overall tracking performance.

The complexity of the scenes was varied during the experiments. In simple cases (`seq5` and `seq7`) objects of similar color to the train were placed along the circular track, some of the objects also had a reflective surface. In other scenes (`seq4` and `seq6`) additionally a large object was placed in the center of the scene resulting in occlusions occurring in all camera images at different times. Finally the camera positions were altered (`seq8`) such that partial and full occlusions occur in two of the cameras, but not in the third. In variations to the basic setup the global lighting of the scene abruptly broke down (`globlight`), dynamic spotlights and shadows were cast (`light`) and total failure of one camera was simulated by holding a hand in front of the lens (`hand`). An excerpt of the sequences can be seen in figures 3 and 4.

In initial experiments different settings for common internal parameters were investigated. These parameters were namely the number of particles and the noise term in the particle filter. As in [4] a particle number of 2000 and a noise term approximately corresponding to the motion speed in the circular movement assumed as ground truth proved to be both computationally manageable and robust in the tracking behavior.

In further experiments the overall performance with different adaptation rates was evaluated. We found a value of $\tau = 0.1$ to give reasonably good results with both fusion mechanism. With higher (i.e. faster) adaptation the smoothening effect on the reliabilities over time seems to be insufficient and tracking results were not robust. Setting the autonomous adaptation too slow or using no adaptation at all the performance of the tracking system completely depended on the choice of the initial weights for the fusion

Sequence	DI	STAPLE a ($D = 8$)	STAPLE a ($D = 32$)	STAPLE b ($D = 8$)	STAPLE b ($D = 32$)
seq4	76.90 (46.18)	354.84 (67.57)	349.52 (85.85)	379.90 (95.27)	340.27 (84.62)
seq6_hand	75.07 (31.91)	383.44 (140.88)	410.92 (184.24)	344.70 (169.44)	387.68 (193.61)
seq6_globlight	86.35 (36.21)	357.12 (111.78)	340.19 (119.85)	336.15 (99.53)	293.83 (128.06)
seq6_light	66.79 (28.26)	406.61 (187.62)	291.64 (178.43)	367.09 (242.45)	373.51 (197.72)
seq5	77.41 (29.52)	82.59 (25.89)	90.36 (24.14)	81.38 (26.59)	87.10 (25.61)
seq7_hand	54.50 (19.90)	128.02 (97.80)	190.87 (162.92)	91.32 (49.23)	316.04 (192.93)
seq7_globlight	59.46 (17.19)	62.82 (14.48)	70.39 (17.44)	66.05 (18.46)	65.53 (15.96)
seq7_light	63.67 (24.32)	64.68 (19.49)	73.49 (27.60)	67.81 (20.73)	68.56 (24.84)
seq8	79.05 (32.45)	78.90 (30.22)	85.17 (29.50)	80.81 (33.15)	79.15 (30.56)
	71.02 (29.55)	213.22 (77.30)	211.39 (92.22)	201.69 (83.87)	223.52 (99.32)

Table 1 Average position estimation errors in mm for the different data fusion approaches with the standard deviations in parentheses.

cues. Some choices might yield better results for specific sequences, our goal however is not to create a system fitting to one specific situation, but a self-organizing system adapting to any given situation.

Finally we directly compared the approaches of Democratic Integration and STAPLE fusion. As a quality measure for Democratic Integration we used a correlation measure. The estimation of sensitivities and specificities in the STAPLE algorithm was performed with half open intervals as described in section 4.2 with different numbers of estimation intervals D. In case a, no interpolation was used, whereas in case b a simple linear interpolation between the interval centers was applied.

As seen in table 1 we achieved typical 3d localization errors between $55mm$ and $70mm$ with both fusion algorithms. The tracked toy train is approximately a box of $100mm$ length, most position estimations therefore lie within the object. Note we also got reasonable results for the scenes with sensor failures (`hand`), as illustrated by figure 4 as well. With the handling and recovery of such errors by adapting the individual influences on the global result, the system can be seen as self-healing.

As expected however the STAPLE fusion is severely affected by many outliers in the scenes with many occlusions (i.e. `seq4` and `seq6`). Breakdowns are the consequence with the tracked train being lost in the scene clutter. Yet for the easier scenes the performance of STAPLE is at least comparable to that of Democratic Integration.

For scenes with continuous successful tracking (i.e. `seq5`, `seq7_globlight`, `seq7_light` and `seq8`) we typically observed a slightly lower standard deviation of the estimation error with STAPLE fusion. This increased tracking accuracy reflects the prediction of more decisive fusion steps resulting in a higher concentration of states with high saliency in one place.

To have an overview of typical processing times we compared the two fusion approaches in this respect as well. We used the four input cues mentioned in section 2, three cameras and a sequence of 10 sec. total length. As a test platform a Pentium 4 with 3.4 GHz and 2 GB RAM was used, a reasonable standard workstation. It can be seen from table 2 that the STAPLE approach is computationally slightly more complex. The difference is only marginal however and both approaches can be run in real-time with an image size of 80x60 pixels. Such small images were also used for the other experiments and still provide a good basis for our object tracking system.

	DI	STAPLE
80x60	8.21s	9.27s
320x240	68.04s	87.55s

Table 2 Average processing times for 10 sec. of video data from three cameras with different image sizes and the different data fusion approaches.

Concluding the experimental results basically a good performance of the overall tracking system could be observed with both fusion systems, although the difficulties in the test sequences include occlusions, reflections, spotlights and a non-uniform object to track. Higher estimation errors occur for scenes with many occlusions, as expected. The higher sensitivity of STAPLE fusion to outliers, which was theoretically predicted before, can be observed in the complex setups with many regular occlusions, especially `seq4` and `seq6`. With the Democratic Integration approach the adaptation mechanisms prevent any drastical errors in these cases as well.

6 Further Work

Several open ends have been mentioned throughout the work. First of all the estimation of the sensitivities and specificities, i.e. the decision error model of the STAPLE algorithm could be improved. This could provide a generally more robust fusion step, but through the iterative feedback positive effects on the adaptation step can be expected as well. The situation of dominating outliers also plays a major role when defining a more appropriate estimation of the reliabilities.

A systematic investigation of hierarchies different than the one depicted in figure 1 might also give further insights

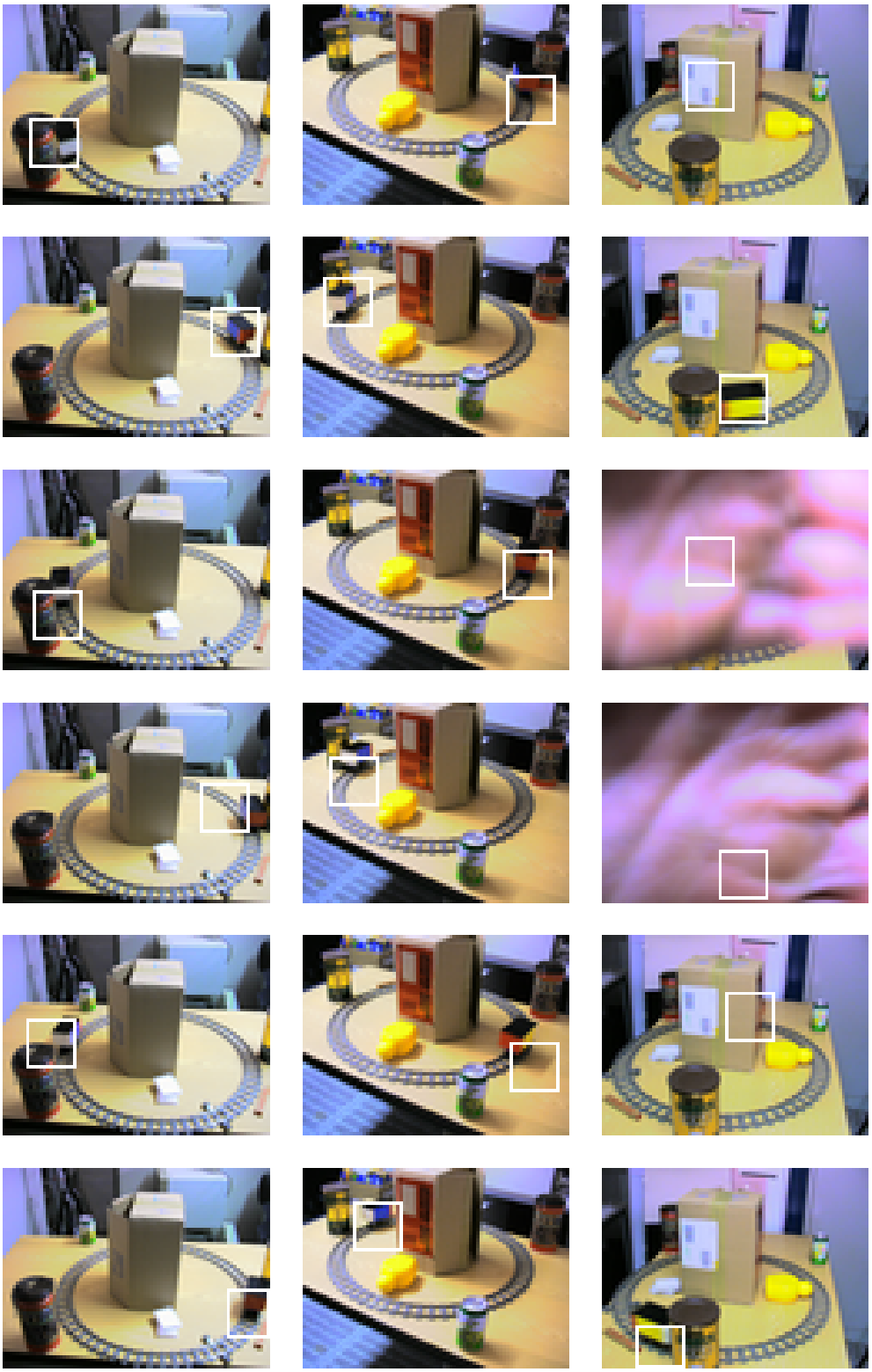

Figure 4 Excerpts from the test sequence `seq6_hand`. Three cameras observe the toy train moving through a complex scene with regular occlusions and a sensor failure simulated by holding a hand in front of one of the lenses. The estimated 3d position of the train is backprojected into the images and indicated by the rectangles.

into the process of data fusion. The potential lying in the correct ordering of several fusion steps was shown in [4], where the hierarchical structure and a flat fusion were compared. Hardly any research has focused on this basic issue in the past.
The need for a camera calibration step prior to performing the data fusion is another starting point for further research. Especially if the sensors do not survey the same area of a scene it is hard to establish a common 3d coordinate system and to compute the projections needed by the presented framework. In such a case it might be useful to run the data fusion independently in the individual cameras and then establish the world coordinate system by combined evaluation of these results.

7 Conclusions

We have presented a general and flexible framework for data fusion. A major concern was adaptivity and providing means for an autonomous self-organization. The adaptation process also provides self-healing mechanisms as they are ubiquitous in biological systems [5], in the sense of detecting and recovering from sensor errors.
Within the framework fusion is not only possible for data of the same state space but using projections, or more generally a mapping between different state spaces, such boundaries can be crossed as well. The framework was constructed with the application of 3d object tracking in mind. The different state spaces in this domain were 2d image coordinates and 3d world coordinates. The framework also permitted an hierarchical approach with a first fusion step on image level and a second step for fusion of the individual cameras.
Two methods to perform the data fusion were then introduced. The first approach of Democratic Integration met all of the requirements to be a self-organizing, robust algorithm. However the lack of a mathematical foundation to the approach gave reason for further research in the area.
The other presented algorithm was an extension of the STAPLE-Fusion. Although further work on its details may be necessary, this method was equally autonomous as the Democratic Integration approach. As an advantage a mathematical basis to the formulas was given by the Bayes equation.
A detailed comparison of the two methods was given afterwards. Both perform very well on a set of complex test sequences, with STAPLE being more sensitive to outliers. Also both algorithms run in real-time if the image size is reduced. The expected self-healing capabilities, i.e. robust behavior and recovery in case of changing environment conditions or failure of a sensor were also shown in the experiments. The observed results also were predicted in a short theoretical analysis of the mathematics behind the data fusion algorithms.
All of the presented methods were applied to the case of 3d object tracking but can be transferred to different applications where a robust, self-organizing data fusion is needed.

8 Acknowledgment

This work was supported by DAAD/NSF grant D/0247202 "Probabilistic Cue Integration of Multimodal Sensor Data in Biologically Inspired Machine Vision Systems". Only the authors are responsible for the content. We like to thank Torsten Rohlfing, SRI International, Neuroscience Program for the discussion on applications of STAPLE.

9 References

[1] Brooks, R.R., Iyengar, S.S.: Real-time distributed sensor fusion for time critical sensor readings. Optical Engineering **36** (1997) 767–779

[2] Sanderson, C., Paliwal, K.K.: Information fusion and person verification using speech and face information. IDIAP Research Report 02-33 (2002)

[3] Denzler, J., Zobel, M., Triesch, J.: Probabilistic Integration of Cues From Multiple Cameras. In Würtz, R., ed.: Dynamic Perception. (2002) 309–314

[4] Kähler, O., Denzler, J., Triesch, J.: Hierarchical sensor data fusion by probabilistic cue integration for robust 3–d object tracking. In: Proceedings of the 6th IEEE Southwest Symposium on Image Analysis and Interpretation. (2004) 216–220

[5] Triesch, J., von der Malsburg, C.: Democratic integration: Self-organized integration of adaptive cues. Neural Computation **13** (2001) 2049–2074

[6] Warfield, S.K., Zou, K.H., Wells, W.M.: Validation of Image Segmentation and Expert Qualitiy with an Expectation-Maximization Algorithm. In: Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention. (2002)

[7] Rohlfing, T., Russakoff, D.B., Maurer, C.R.: Performance-Based Classifier Combination in Atlas-Based Image Segmentation Using EM Parameter Estimation. IEEE Transactions on Medical Imaging **23** (2004) 983–994

[8] Isard, M., Blake, A.: Condensation – conditional density propagation for visual tracking. International Journal of Computer Vision **29** (1998) 5–28

[9] Tsai, R.: A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal of Robotics and Automation **3** (1987) 323–344