Using Particle Filters for Autonomous Underwater Cable Tracking ⋆

Stephan Wirth * **Alberto Ortiz** ** **Dietrich Paulus** *
Gabriel Oliver **

* *University of Koblenz and Landau, Koblenz, Germany (e-mail: {stwirth | paulus}@uni-koblenz.de)*
** *University of the Balearic Islands, Palma de Mallorca, Spain (e-mail: {alberto.ortiz | goliver}@uib.es)*

Abstract: This paper describes the use of particle filters for visual tracking of underwater elongated structures such as cables or pipes. Models for probabilistic tracking are obtained directly from real underwater image sequences. Extensive experiments on realistic underwater videos of power cables installed almost 30 years go (i.e. they cannot be easily discriminated from the background) show the robustness and performance of this approach.

Keywords: Cable Tracking, Particle Filters, Autonomous Underwater Vehicles (AUV), Robot Navigation, Robot Vision

1. INTRODUCTION

The automatic estimation of the pose (i.e. the position and the orientation) of subsea elongated structures such as cables or pipes from camera images is the first step towards autonomous visual inspection by means of underwater vehicles. Distracting background, such as rocks or algae growing on top and nearby cables/pipes, complicate however their detection (see Fig. 1). For instance, the determination of the cable pose via border detection followed by line extraction as used in (Balasuriya *et al.*, 1997; Balasuriya and Ura, 1999) tends to fail under those circumstances. Other approaches use mainly segmentation (Antich and Ortiz, 2003) or texture descriptors (Grau *et al.*, 1998). In all these approaches, ambiguities may occur when rocks or marine growth form shapes and textures that resemble a cable/pipe. In this case, deterministic approaches which try to find out the exact cable/pipe pose can fail. Limitations to search in regions of interest (ROI), which are determined from processing results of previous frames, address this problem and solve it for most situations. However, if ambiguities occur within these ROIs, the approach still tends to fail.

In these cases, stochastic solutions must therefore be superior to deterministic approaches. Besides, through modeling probabilities it is possible to make assumptions on the state of the object of interest even if the number of extracted features is small.

This work introduces a new tracking system for elongated structures appearing along image sequences, which is based on particle filters (see e.g. Isard and Blake (1998); Arulampalam *et al.* (2002)). The main interest on particle filters is their natural ability to model multi-dimensional multi-modal probability density functions, what makes them of more general application, although at the expense of a larger computational cost.

Along the paper, a model for cables as they appear in undersea images is developed within this probabilistic framework. Furthermore, real images from inspection runs captured by remotely operated vehicles (ROVs) are used to find out an appropriate movement model which describes the changes of the cable parameters over time. Finally, an observation model which detects edges by a derivative of a Gaussian filter is also defined. All these models are combined in a particle filter which sequentially estimates the likelihood of the cable pose. For every frame in the video sequence, the previously computed probability density function of the cable parameters is used to predict – via application of the movement model – the cable pose in the next frame. Then, the probability density function is updated by means of the observation model. The most likely cable pose is finally determined from the resulting density.

The paper is organized as follows: section 2 gives a brief introduction to particle filters; models for probabilistic cable tracking are developed in section 3; section 5 outlines our experiments with real underwater image sequences and conclusions can be found in section 6.

2. PARTICLE FILTERS

In particle filters, probability density functions are approximated by a set of N weighted *particles*, the sample set $S = \{(\mathbf{s}^{(n)}, \pi^{(n)}) | n = 1, \ldots, N\}$. Each particle represents a particular (hypothetical) configuration $\mathbf{s}$ of the variables (i.e. one state) in the state space, together with an importance weight π, where the state space is chosen for the given application. The weights $\pi^{(n)}$, as they describe a probability density function, sum up to one: $\sum_{n=1}^{N} \pi^{(n)} = 1$. The evolution of the sample set is defined by two models: the *movement model*, which defines how

⋆ This project was partly supported by the EU program Socrates/Erasmus as well as by spanish CICYT and FEDER funds (project number DPI2005-09001-C03-02)

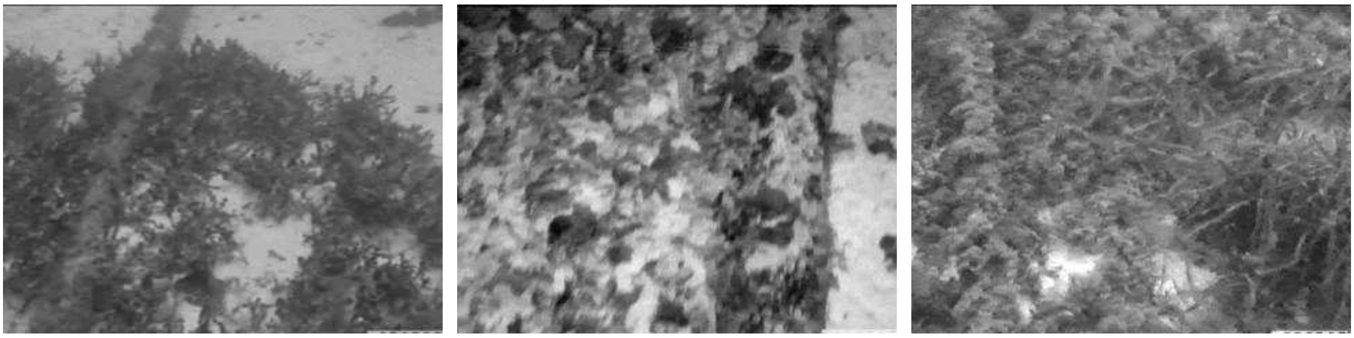


Fig. 1. Images of underwater cables in which the cable is difficult to detect.

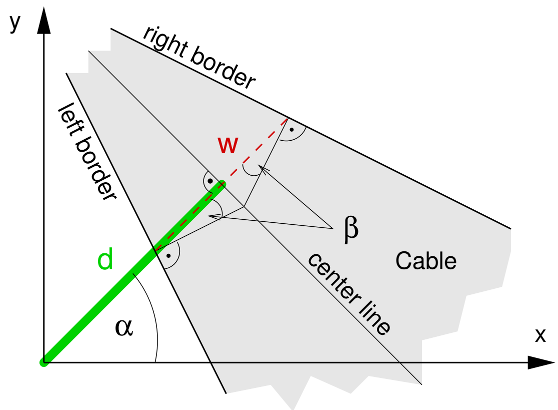


Fig. 2. Cable model. The coordinate system origin is at the center of the image.

the particles are moved in state space from one time step to the next, and the *observation model*, which is used to weight the particles according to a given observation (e.g. a camera image). To keep the focus on the modes of the probability density function, we use the following resampling method: whenever the (estimated) number of effective particles

$$\widehat{N_{\text{eff}}} = \frac{1}{\sum_{n=1}^{N} (\pi^{(n)})^2} \tag{1}$$

falls below $N/2$, a new particle set is drawn from the old set by choosing a particular sample i with probability $\pi^{(i)}$. In this way, samples with higher probabilities are multiplied and samples with less importance are discarded. At any time t, the estimated mean state $\hat{\mathbf{x}}_t$ can be computed by

$$\hat{\mathbf{x}}_t = \sum_{i=1}^{N} \pi_t^{(i)} \mathbf{s}_t^{(i)} \tag{2}$$

3. MODELS FOR CABLE TRACKING

3.1 Cable Model

Figure 2 illustrates our model of an undersea cable as it appears in camera images. The chosen parameters are listed in table 1. While tracking a cable, the underwater vehicle is supposed to navigate at a constant distance to the seabed. Therefore, the vehicle movements can be confined within a 2D plane parallel to the ground. In this way, the vehicle only changes its yaw angle and performs longitudinal translations. On the other hand, the camera is

Table 1. **Cable model parameters.**

Parameter	***Description***
d	Distance from image center to cable centerline
α	Angle of cable centerline
w	Cable width
β	Cable "skew", arises from perspective distortion if the camera image plane is not parallel to the cable main axis

heading downwards and to the front. Consequently, since the curvature of the cable is very small, the cable borders appear as straight lines in the image.

3.2 Movement Model

Considering the above-mentioned constraints, we can make the following assumptions on the cable state parameters:

- Neither the cable "skew" β nor the cable width w change significantly over time.
- $0° \leq \beta \ll 180°$.
- The cable angle α changes smoothly when the vehicle rotates around its yaw axis.
- The cable distance d changes when $\alpha \neq 0°$ and the vehicle is moving.

Let $\mathbf{x}_t = (d_t, \alpha_t, w_t, \beta_t)^T$ denote the correct cable state at time t. The state $\mathbf{x}_{t+1}$ of the next time step $t+1$ can then be expressed by

$$\mathbf{x}_{t+1} = \mathbf{x}_t + (\Delta\mathbf{x})_t = \begin{pmatrix} d \\ \alpha \\ w \\ \beta \end{pmatrix}_t + \begin{pmatrix} \Delta d \\ \Delta\alpha \\ \Delta w \\ \Delta\beta \end{pmatrix}_t \quad . \tag{3}$$

As already mentioned above, w and β do not change significantly over time, hence Δw and $\Delta\beta$ are very small and can be modeled as Gaussian random noise. The other two increments can be bigger but they change smoothly over time: $(\Delta d)_{t+1} \approx (\Delta d)_t$ and $(\Delta\alpha)_{t+1} \approx (\Delta\alpha)_t$. We model these increments by including an instant rotation velocity v_α and an instant translation velocity v_d into the state:

$$\mathbf{x}_t = \begin{pmatrix} d \\ \alpha \\ w \\ \beta \\ v_d \\ v_\alpha \end{pmatrix}_t \quad . \tag{4}$$

Fig. 3. Application of the edge filter.

The overall transition of a state $\mathbf{x}$ from time t to $t+1$ is then defined as

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \begin{pmatrix} v_d \\ v_\alpha \\ \Delta w \\ \Delta\beta \\ \Delta v_d \\ \Delta v_\alpha \end{pmatrix}_t \quad . \tag{5}$$

Here again Δv_d and Δv_α tend to be small and can be modeled as Gaussian random noise. We apply this model to a set of particles which estimates the real cable state. In each time step, the sample state $\mathbf{s}^{(i)}$ of each particle i is modified:

$$\mathbf{s}_{t+1}^{(i)} = \mathbf{s}_t^{(i)} + \begin{pmatrix} v_d \\ v_\alpha \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}_t^{(i)} + \begin{pmatrix} G(\sigma_d^2) \\ G(\sigma_\alpha^2) \\ G(\sigma_w^2) \\ G(\sigma_\beta^2) \\ G(\sigma_{v_d}^2) \\ G(\sigma_{v_\alpha}^2) \end{pmatrix} \tag{6}$$

with $G(\sigma^2)$ being a function that returns an $\mathcal{N}(0,\sigma^2)$ distributed Gaussian random number. The variances σ_d^2, σ_α^2, σ_w^2 and σ_β^2 model process and measurement noises while $\sigma_{v_d}^2$ and $\sigma_{v_\alpha}^2$ also have to compensate velocity changes (i.e. accelerations). In particle filters terminology, the addition of the instant velocities is called *drift* and the addition of the random noise is the *diffusion*.

Note that this model assumes that all cable parameters are independent, which does not need to be the case. However, experiments have shown that this simple model is precise enough to follow cable movements.

3.3 Observation Model

In every cycle of the particle filter, each particle has to be weighted according to the probability of its state. Therefore we use the cable pose which is stored in a particle to project the (hypothetical) cable borders onto the current camera image. Equally spaced along these borders we apply a derivative of a Gaussian filter with its direction perpendicular to the border (see Fig. 3).

For each border, its filter responses are summed up to form the "border score". The cable may appear bright on dark ground or vice versa. Therefore we check if the border scores from left and right border have opposite signs. If so, the total score for the checked cable state hypothesis is the sum of the absolute values of left and right border scores. Otherwise, it is set to a very small value.

This total score is then stored as the weight of the particle. After every particle has been weighted, their weights are normalized to ensure that they sum up to one.

4. THE ALGORITHM

The entire cable tracking algorithm consists in the straightforward application of the previously described models. In every iteration, the input of the algorithm is the particle set of the previous iteration together with a new observation. The output is the resampled, moved, and newly weighted particle set, from which the current cable state can be estimated. The single steps of the algorithm are the following.

Let $\mathbf{S}_t = \{(\mathbf{s}_t^{(i)}, \pi_t^{(i)}), i = 1,\ldots,N\}$ be the set of N weighted samples (particles) at time t.

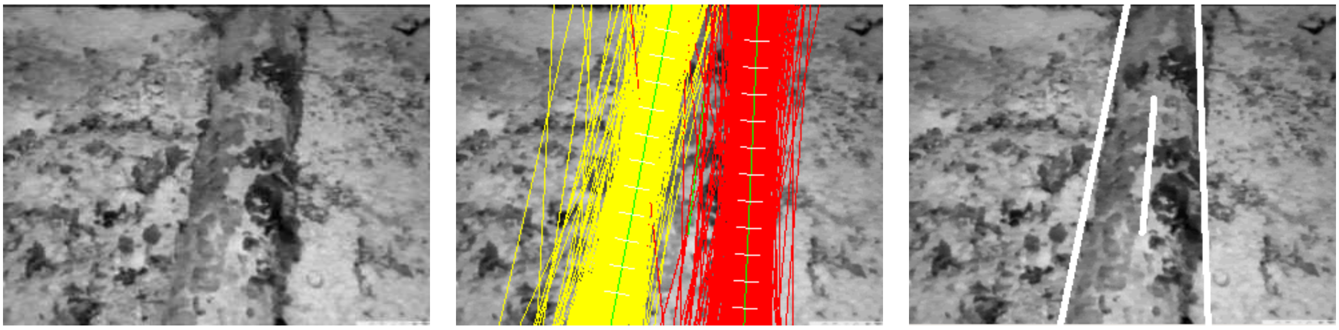
(1) *Initialization:* to create the initial particle set
$$\mathbf{S}_0 = \{(\mathbf{s}_0^{(i)}, \pi_0^{(i)}), i = 1,\ldots,N\}$$
either use ground truth knowledge or randomize the particles over an adequate space with equal weights $\pi_0^{(i)} = 1/N$.

(2) *Iterate:* do the following steps to let the particle set track the state density for time t
   - (a) *Resampling:* if $\widehat{N_{eff}} < N/2$, resample N new particles from $\mathbf{S}_{t-1}$ to create the new (not yet weighted) sample set
   $$\mathbf{S}'_{t-1} = \{\mathbf{s}_{t-1}^{(i)\,\prime}, i = 1,\ldots,N\}$$
   as described in section 2.
   - (b) *Drift and Diffuse:* update the particle states according to the cable movement model (refer to section 3.2) to form the (not yet weighted) sample set $\mathbf{S}'_t$.
   - (c) *Measure:* use the currently observed camera image to update the observation model and use this model to calculate the new particle weights $\pi_t^{(i)}$ as described in section 3.3. Now the particle set $\mathbf{S}_t = \{(\mathbf{s}_t^{(i)}, \pi_t^{(i)}), i = 1,\ldots,N\}$ approximates the posterior density.
   - (d) *Estimate:* use $\mathbf{S}_t$ to estimate the cable state (refer to section 2)
   $$\hat{\mathbf{x}}_t = \sum_{i=1}^{N} \pi_t^{(i)} \mathbf{s}_t^{(i)} \,.$$
   This estimation can now be used as input for the AUV controller.

Figure 4 illustrates how the cable state is estimated through testing a set of hypotheses. The plots of Fig. 5 show how the set of particles approximate the probability density function for the cable angle and the cable distance.

5. EXPERIMENTS

The proposed algorithm was applied on real underwater image sequences (320×240 pixels), which were taken by



(a) (b) (c)

Fig. 4. (a) shows a typical image of a submarine cable; (b) shows the 500 hypothetical cable poses that are modeled by the particle filter for that image (the short white lines mark the filter windows for an exemplary particle); (c) estimated cable pose.

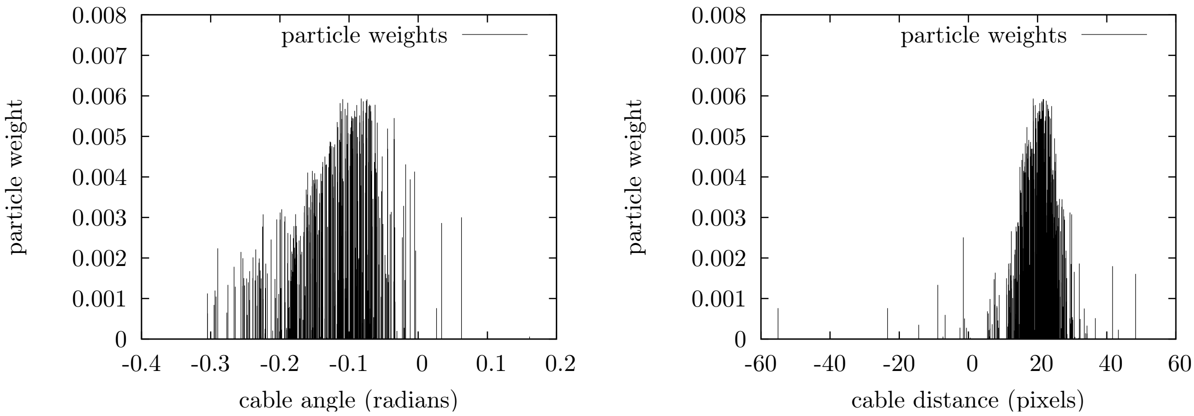


Fig. 5. Cable angle and cable distance densities. Each line corresponds to one particle. Note that the plots show only two of the six dimensions of the approximated probability density function.

ROVs and whose complexity was similar to the one exhibited by the images of Fig. 1. The particle filter was configured to define and handle between 300 and 500 particles. The variances for the movement model were determined through interactive experiments. The observation model made use of filters of 11×1 pixels every 20 pixels along the cable borders. The initialization of the particle distribution was based on hand-labeled ground truth.

For illustration purposes, and among the different successful experiments carried out, Fig. 6, 7 and 8 compare estimated cable distance and angle with ground truth for three image sequences. As can be observed, the estimated values are very close to the ground truth in every case. Besides, Fig. 9 shows the projections of the estimated cable state on frames of another video sequence. The average computation time on a 1.5 GHz Pentium PC was about 16 ms per filter cycle, which corresponds to a rate of approximately 62 frames per second.

6. CONCLUSION

A probabilistic approach for visual cable tracking based on particle filters has been proposed. Models for appearance and movement of cables in image sequences as well as a measurement function have been developed. Taking into account the performance and robustness of the new tracker, which have been demonstrated through experiments, we conclude that the approach is suitable for online cable following and inspection tasks even in situations where the cable is hardly visible.

REFERENCES

Antich, J. and A. Ortiz (2003). Underwater cable tracking by visual feedback. In: *IbPRIA* (A.C. Campilho, N. Perez, A. Sanfeliu and F.J. Perales, Eds.). Vol. 2652 of *LNCS*. Springer. pp. 53–61.

Arulampalam, S., S. Maskell, N. Gordon and T. Clapp (2002). A tutorial on particle filters for on-line non-linear/non-Gaussian bayesian tracking. *IEEE Transactions on Signal Processing* **50**(2), 174–188.

Balasuriya, B.A.A.P. and T. Ura (1999). Multi-sensor fusion for autonomous underwater cable tracking. In: *OCEANS '99 MTS/IEEE Conference Proceedings*. Vol. 1. Seattle, WA, USA. pp. 209–215.

Balasuriya, B.A.A.P., M. Takai, W.C. Lam, T. Ura and Y. Kuroda (1997). Vision based autonomous underwater vehicle navigation: Underwater cable tracking. In: *OCEANS '97 MTS/IEEE Conference Proceedings*. Vol. 2. Halifax, NS, Canada. pp. 1418–1424.

Grau, A., J. Climent and J. Aranda (1998). Real-time architecture for cable tracking using texture descriptors. In: *OCEANS '98 Conference Proceedings*. number 3. Nice, France. pp. 1496–1500.

Isard, M. and A. Blake (1998). Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision* **29**(1), 5–28.

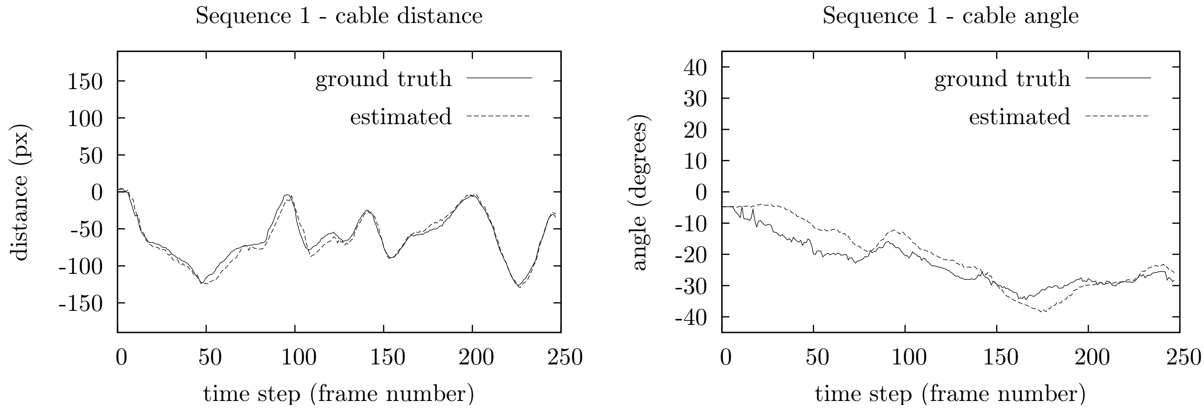


Fig. 6. Experiment on an image sequence with 248 frames. The plots compare (hand-labeled) ground truth with computation results.

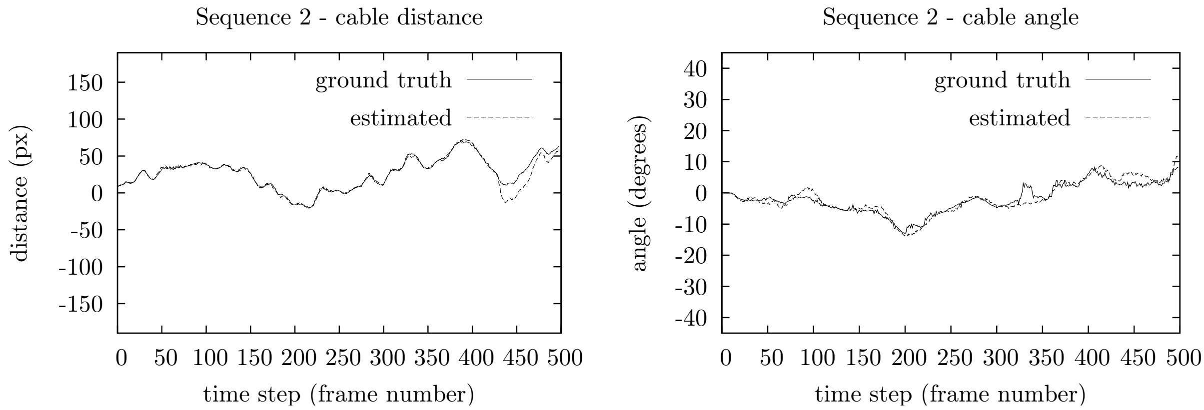


Fig. 7. Experiment on an image sequence with 498 frames.

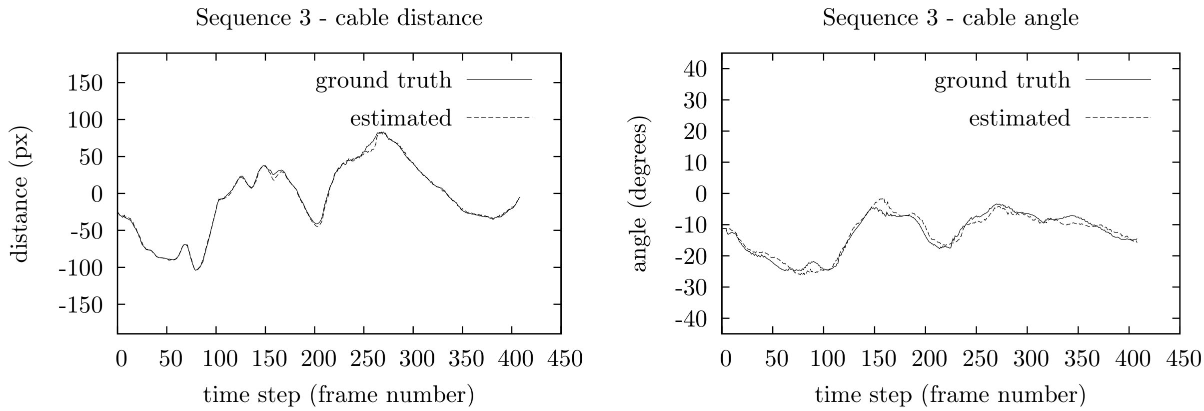


Fig. 8. Computation results of another experiment on an image sequence with 409 frames.

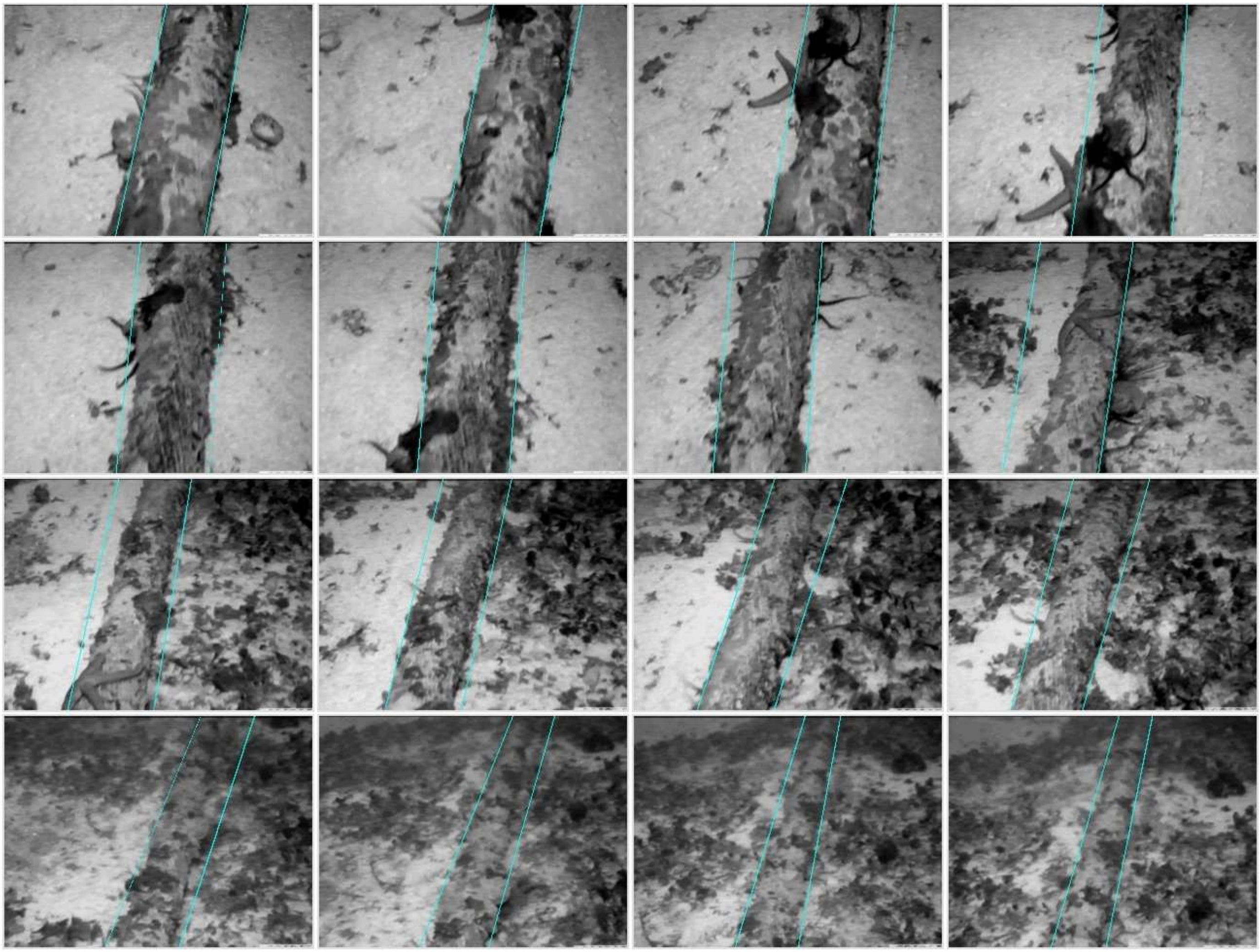


Fig. 9. Estimated cable states, projected on the camera images. Every tenth frame of the sequence is shown. Though the fast change of the cable width between frame 60 and 70 (7th and 8th image) cannot be followed by the proposed movement model, the correct cable state is recovered after a couple of frames.