Image-Based Supervision of a Periodically Working Machine

Mario Frank[1], **Fred A. Hamprecht**[2]

[1] Department of Computer Science, ETH Zurich, 8092 Zurich, Switzerland
[2] Heidelberg Collaboratory for Image Processing, University of Heidelberg, 69120 Heidelberg, Germany

Abstract Most industrial robots perform a periodically repeating choreography. Our aim is to detect disturbances of such a periodic process by a visual inspection system that can be trained with a minimum of human effort and interaction. We present a solution that monitors the robot with a time-of-flight 3D camera. Our system can be trained using a few unperturbed cycles of the periodic process. More specifically, principal components are used to find a low-dimensional approximation of each frame, and a One-Class Support Vector Machine is used for one-class learning. We propose a novel scheme for automatic parameter tuning, which exploits the fact that successive images of the training class should be close in feature space. We present exemplary results for a miniature robot setup. The proposed strategy does not require prior information on the dimensions of the machine or its maneuvering range. The entire system is appearance-based and hence does not need access to the robot's internal coordinates.

1 Introduction

Industrial environments require reliable security solutions for an accident-free cooperation of humans and autonomously working machines. The simplest, and least flexible, solution is to separate man and machine by static safety margins such that any space potentially occupied by the machine cannot be entered by humans. These boundaries can either be of a physical nature or can, for instance, be guarded by light curtains which raise an alarm or stop the machine when trespassed. Recent approaches employing both local and global sensors range in complexity from mere intrusion detection to full-fledged human-robot cooperation: A "skin" of haptic sensors for collision detection is investigated in [1,2]. In [3,4], a wrist mounted laser scanner is introduced which allows to monitor the proximity between man and robot. Algorithms for collision avoidance using such a scanner have been proposed in [5].

Global sensors, in contrast to local sensors, monitor the entire machine workspace. The challenge with such an approach is to be sensitive to obstacles in the working path while tolerating the actions of the robot itself as well as varying environmental conditions, e.g. illumination. An early attempt to the automatic detection of humans in video data was presented in [6]. Features such as height, width and area (in pixel space) are extracted from the object shape which is obtained by a difference-image method. The classifier consists of hyper-rectangles (thresholds) in this feature space. The methods proposed in [7,8] use skin color to discriminate humans from other moving objects in sequences acquired with video cameras. In [9], a backprojection method is proposed that reconstructs the positions of obstacles in the working space in world coordinates from multiple images. In [10], stationary cameras are used as passive light barriers. The scene is segmented into a variable foreground and supposedly constant background, and changes in the background are detected and interpreted as violations of the working space. In [11,12], difference-image methods with multiple cameras as proposed in [13] are used to detect obstacles and to enable path planning. A method for monitoring predefined areas of interest with a gray-scale camera is presented in [15]. The task of obstacle detection or intrusion detection is methodologically closely related to the computer-vision based defect detection task (see [14] for a survey on such approaches) where the input data is often also of sequential type (e.g. in [16]).

Our aim has been to develop an intrusion detection scheme that can be configured with an absolute minimum of human interaction: the device is composed of a depth sensor with spatial and temporal resolution and a computer. The idea is to adjust the sensor's field of view such that it can survey the machine motion to be monitored, but is blinded from *permissible* geometric changes in the environment, e.g. in the background of the machine. For training, the sensor needs to acquire a few *unperturbed* cycles of the machine's scheduled periodic movement; during those cycles, the human operator has to guarantee that no intrusions occur. The device should then configure itself to detect any subsequent deviations from the desired motion pattern, in particular to detect intrusions into the sensor's field of view. Whenever such a deviation oc-
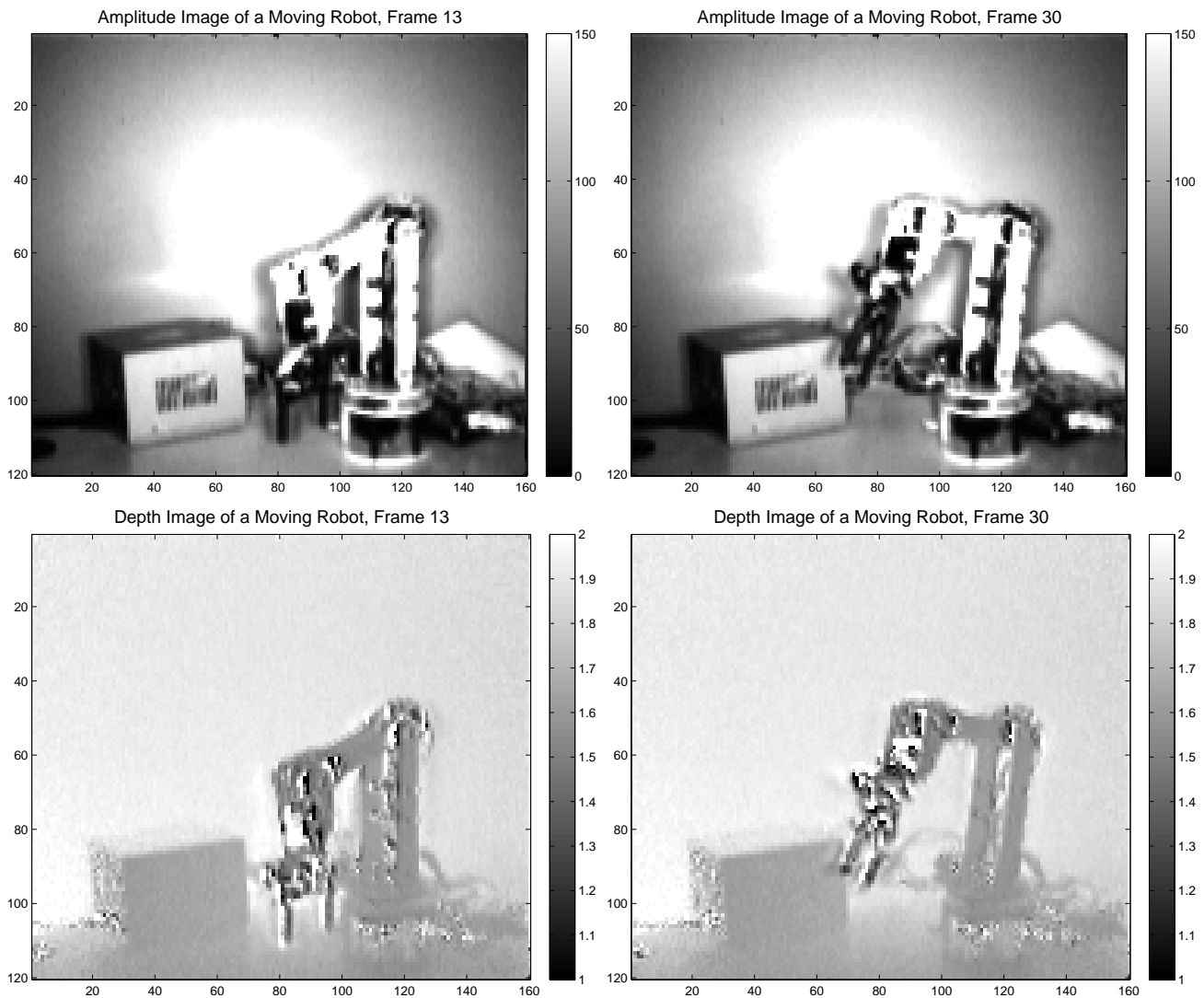
Mario Frank[1], Fred A. Hamprecht[2]

Fig. 1 Two frames (left to right) of a training sequence of unperturbed robot operation. Top row: Amplitude image of backscattered infrared radiation in arbitrary units. Bottom row: Color-coded depth map with shade indicating distance in meters. Noise and motion artifacts at surfaces with low reflectivity (black in the amplitude images) are apparent.

curs, the device will raise an alarm which can, for instance, be used to stop the machine.

Although our approach does not allow for direct human-robot interaction (since it only provides a global alarm and cannot localize obstacles), it works out of the box: It only requires the sensor setup and a single button to initiate and end the training phase, and a channel to pick up and propagate the alarm signal. We present an implementation of our concept using a time-of-flight range camera together with the intrusion detection algorithm sketched in the following. First, we project the high-dimensional sensoric outputs – the depth maps – onto a low-dimensional subspace by a mapping which is learned during the training phase by means of principal component analysis (PCA), cf. Section 2.2. Second, we learn the permissible machine motion pattern in this low-dimensional space using a One-Class Support-Vector Machine (OC-SVM), given observations which are guaranteed to be

free from perturbations (Section 2.3). Finally, the output from the OC-SVM is used to detect deviations from the allowed periodic motion pattern. There, our main contribution is the automated adjustment of all OC-SVM parameters by exploiting a problem-specific connectedness assumption.

2 Sensor and Algorithmic Preliminaries

2.1 The Camera

For image acquisition, we use a continuous-wave time-of-flight (TOF) 3D camera working in the infrared regime [17, 18]. The acquired depth maps have the advantage of being mostly illumination invariant. The camera works as follows: Infrared light modulated with radio frequency is emitted by diodes and backscattered from the scene. Each pixel of the sensor chip detects this light and correlates the signal with the

electronic modulation signal directly after charge generation. From three or more of these correlated signals, the phase shift of the backscattered light and hence the distance can be calculated [17]. The variance of the estimated distance strongly depends on the amplitude of the signal amplitude and is subject to structured noise [19]. We therefore apply adaptive normalized convolution filters [20] for noise reduction. The results reported here are obtained from a prototype (PMD[vision] 19k, PMDtec, Siegen, Germany) which delivers depth maps and amplitude images at a resolution of 160×120 pixels and a frame rate of about 10 Hz.

2.2 Dimensionality Reduction and Motion Trajectory

A $m \times n$ depth map of a 3D scene is represented by a vector in $N = m \cdot n$-dimensional space. A sequence of frames hence corresponds to a manifold in this N-dimensional space, and can be summarized by a vector μ representing the mean image, as well as a matrix X whose columns are given by the vectorized depth images minus μ.
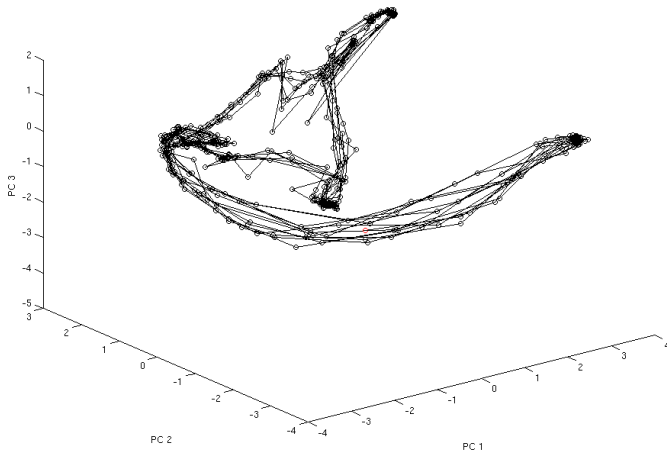

Fig. 2 The training trajectory after projection onto the first three principal components (PCs). The circles represent frames that were actually observed, the line segments connect pairs of consecutive frames. In most cases, the trajectory is more closely sampled in phases in which the robot slows down to change direction. Samples are more sparse in phases in which the robot moves, and hence changes appearance more quickly.

The intrinsic dimensionality of virtually all movies is much smaller than the number of pixels N (as illustrated in Fig. 3); to simplify learning of the repetitive motion pattern, it is advantageous to first map the data to a subspace whose dimensionality is sufficient but not excessive. Numerous nonlinear methods for such a dimensionality reduction have been proposed, e.g. [21–23, 29]; however, the simple and well-known linear method PCA has proven successful in our application scenarios. The left singular vectors of X are the principal components, or "eigenfaces". This set of principal components constitutes a new orthogonal basis for the observations,

and a user-selected subset spans an optimal linear subspace. Fig. 2 shows a movie of a periodically moving machine projected onto this subspace.

If the movie shows a periodic motion, the corresponding trajectory in image space is a closed (but not necessarily circular) loop. Note that this is true for any periodic motion, circular or not: For any scene showing an arbitrary periodic motion, the pixel values of any two frames with a temporal distance of one period are almost equal, and their corresponding points in the full image space or the principal components subspace are thus close to each other (Fig. 2).

The entire trajectory represents the class of permissible geometric configurations of the machine during its action cycle. During the training phase, this manifold can be learned (see next section). Once training is completed, the surveillance phase can start. During surveillance, any new and previously unseen depth map will be projected onto this subspace. If the current depth map is free from undesired perturbations, its projection will end up in the proximity of the learned manifold, indicating that the machine is in a secure state. If the depth map contains an obstacle, the projection will deviate from the learned trajectory, and an alarm can be raised.

2.3 One-Class Support Vector Machines (OC-SVM)

The training phase only yields "negative" examples, i.e. samples from the class of permissible geometric configurations. "Positive" examples, corresponding to process deviations that should be detected by the system, are not available. We are hence faced with a "one-class" learning problem and thus use a One-Class Support Vector Machine (OC-SVM) [24], a relative of standard Support Vector Machines (SVM)[25]. Like in SVMs, OC-SVM linearly separates the data in an implicit nonlinear high-dimensional kernel-space [25]. When projected back to the original feature subspace, the linear decision boundary becomes a non-linear "contour" surface that delimits the support of the normal observations in feature space.

OC-SVM estimates the proximity of a new, unseen observation to the set of negative training examples. For a radial basis function kernel (RBF), one must select two parameters: the width σ of the Gaussian kernel function used to measure the similarity of two observations, and the threshold θ on the proximity. The classification performance depends on this crucial choice. An appropriate kernel width and threshold produces a closed "tube" enclosing the entire trajectory of the training points such that each line segment between two consecutive frames is inside the tube. A too low proximity threshold θ or a too large kernel width σ yields a tube that is too wide and may possibly produce false negatives, i.e. tolerate scenes for which an alarm should be raised. If θ is chosen too high or σ too low, the closed training trajectory becomes fragmented into multiple "islands" and gives rise to false alarms unless a very large training set is available. Based on these considerations, we propose a novel method in Section 3.2 to adjust these important parameters automatically.
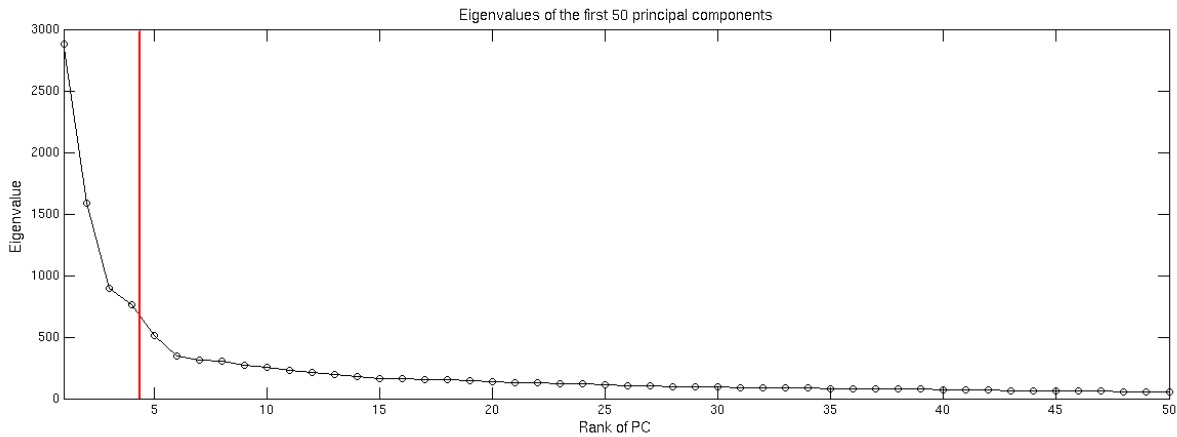
Fig. 3 The eigenvalues of the training sequence drop quickly. Therefore, a low-dimensional subspace spanned by only four eigenvectors (vertical line) suffices to approximate the data. This graph shows only the largest 50 eigenvalues out of 499 nonzero ones.
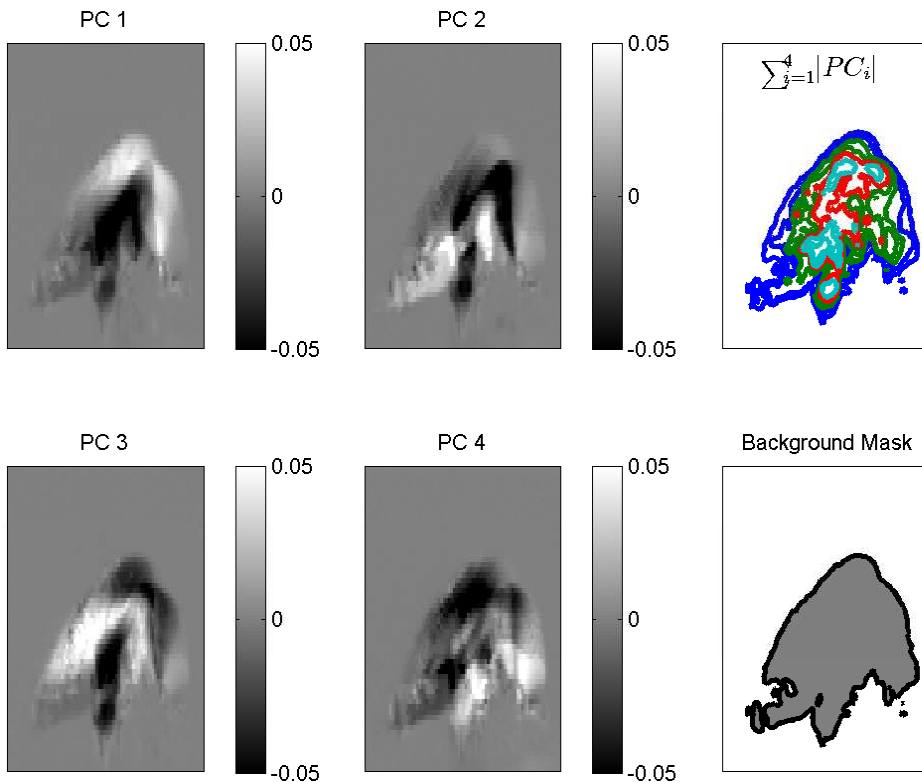

Fig. 4 The first four principal components of a training sequence of undisturbed robot operation suffice to distinguish safe working states from intrusions. The image can, in addition, be segmented into background and operative range by summing up the absolute values of these PCs (contour plot). White pixels indicate background (bottom right).

3 Automated Training under Connectedness Regularization

This section elaborates on a proof-of-principle realization of the proposed concept. We set up a programmable toy robot performing complex periodical motions as the monitoring task. We record a training sequence of 500 frames during unperturbed robot operation, covering approximately ten motion periods. Then we train an OC-SVM classifier on this sequence, which is represented in a subspace spanned by the first few principal components.

3.1 Subspace Configuration

Choice of the Principal Components The eigenvalues of the principal components are dropping quickly such that one can expect to approximate the images sufficiently well by projecting them on a low-dimensional subspace only. The scree

plot in Fig. 3 has a small shoulder at eigenvalues three and four, indicating that they have a similar importance in describing the sequence. In this particular example, we have chosen to take the first four dimensions for training the classifier, covering roughly 40% of the overall variance in the training sequence. This choice is a tradeoff between approximation quality (improving with more principal components) and the difficulty of density estimation / outlier detection (also growing with more principal components for a constant number of observations). The number of principal components is varied for each of the experiments presented below to investigate the influence of this choice.

A method for background detection Pixels always showing a constant scene background, regardless of the current pose of the robot, are never occluded by the moving machine or its parts and will hence be constant, up to noise, during the training phase. Consequently, the first principal components exhibit low amplitude at these pixels, as can be seen from Fig. 4. When a frame is projected to these first principal components only, the low-amplitude regions of the basis functions will cancel all deviations from the mean image in those regions. This masks out all events that are happening within the field of view, but beyond the operative range of the robot. These pixels are still of interest, however, because an approaching invader should be detected before he or she reaches the actual maneuvering range, giving time to e.g. slow down or stop the machine.

The detection of such situations with imminent risk of collision formulates an additional surveillance task which is independent of the surveillance of the operation space. It requires a segmentation between the area where the robot moves (the maneuvering range) and the adjacent neighborhood. We use the thresholded sum over the absolute values of the first few principal components (see Fig.4, top right) to define an auxiliary mask that comprises all pixels within the maneuvering range of the machine. An additional basis function (Fig.4, bottom right) can finally be obtained as the complement of this mask: it has nil values inside the maneuvering range and one outside. We will refer to this mask as the background detector. All projections of "normal" images onto this auxiliary basis function lead to values varying slightly around zero. However, if there is an object close to the operative range, the projection yields a larger value. To classify such events, one simply has to apply a threshold that copes with small random deviations from zero due to noise. We automatically determine this threshold using the following heuristic. First, we compute the variance of the projection of the negative training images onto the background detector. Then, we set the threshold twice as high as this variance.

3.2 Automated OC-SVM Configuration

For classification, we use a One-Class SVM with a Gaussian RBF kernel from the publicly available SPIDER packet [28]. With such a kernel, the training samples are always linearly

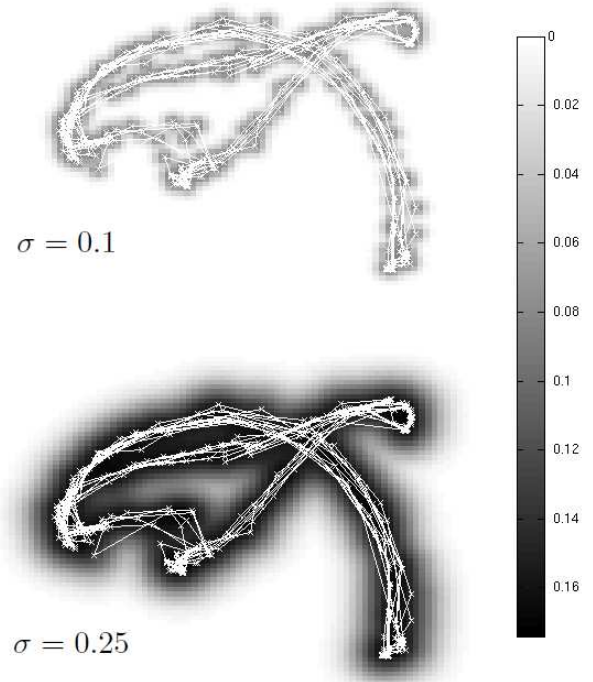Support of a Projection on the first two Principal Components

$\sigma = 0.1$

$\sigma = 0.25$

Fig. 5 Proximity to a training sequence that was projected onto the first two principal components. The crosses are training samples interpolated with white lines. Top: If the kernel width is chosen too small, areas between temporally adjacent trajectory points are flagged as outliers: the trajectory becomes fragmented. Bottom: Appropriate choice of the kernel width. Applying a threshold on the proximity creates a "tube" partitioning the feature space into acceptable and outlying states.

separable from the origin in the corresponding feature space since they are all lying on a sphere within the first orthant as shown in [24].

The RBF kernel width σ determines the nature of the nonlinear projection to a higher-dimensional space that is implicitly performed by the OC-SVM. Its choice is crucial: it scales the similarities of observations in the implicit high-dimensional kernel space and thereby determines how fine-grained the proximity estimation back in the principal component subspace becomes. For large σ, the "tube" becomes smoother and larger; in the extreme case, a single hypersphere comprising all negative training examples results. For small σ, the tube will fragment into islands, resulting in the extreme case in a small hypersphere around every single training example (see Fig. 5).

With this in mind, σ can be considered as a regularization parameter that – together with the proximity threshold θ – influences the shape and bore of the tube enclosing the trajectory of the training sequence in the input space and, more importantly, determines if this tube is connected. The parameters σ and θ must be chosen such that all true negative events are inside the tube and all true positives outside.

The proximity threshold θ (see Section 2.3) can be chosen automatically in a fast and convenient manner: since the proximity drops rapidly at a short distance from the training points (see Fig.5), we select the threshold to be ten times smaller than the smallest proximity that one of the training samples has to the others, as measured by the OC-SVM (the proximity is $\in [0, 1]$ where 1 means that a sample coincides with another). In Fig. 6 the θ parameter that has been selected in such a way is plotted as a dashed line.
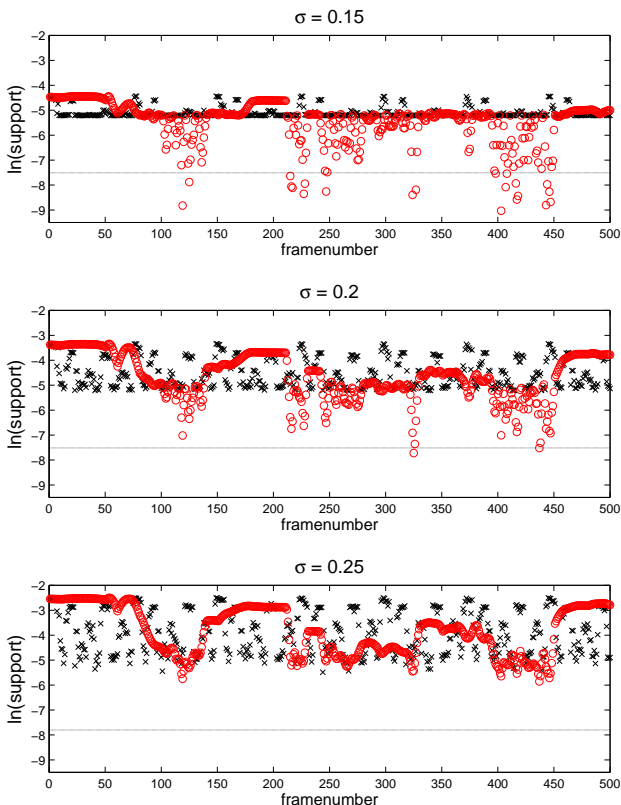

Fig. 6 Logarithmic proximity of points in PC subspace, as measured by OC-SVM with three different kernel widths σ. The crosses indicate points from the negative training class (the actually observed frames), the circles are linear interpolations in PC subspace between consecutive pairwise training samples. The dashed line represents the proximity threshold θ that discriminates between the training class and the complementing space. θ is always chosen ten times smaller than the smallest proximity measure of the training samples. The algorithm increases the kernel width σ until the proximity measure of each interpolated point is higher than θ.

We select the second parameter, the kernel width σ, in the following way. We exploit the fact that the training samples have a temporal ordering and are not just independently sampled from a static distribution. If the sampling rate of the sensor were infinitely high, the training sequence would thus provide a connected line in PC subspace. This connectedness assumption implies that, for an adequate sampling rate, all points on the line segment between any two consecutive

training examples in PC subspace should also belong to the negative class of permissible machine states.

Therefore we propose to regularize the SVM by in the following algorithmic procedure: First, train the classifier on the input data using a very small initial kernel width σ^0 resulting in a proximity landscape that rapidly decays away from the individual training samples. The proximity threshold θ which is, for each σ, chosen to be one order of magnitude smaller than the smallest proximity at any training point, partitions the feature space into complementary regions of large proximity (negative prediction) and low proximity (positive prediction, alarm). Then, iteratively increase the kernel width until all points on the line segments between any two consecutive training examples end up inside the proximity boundary defined by θ. Any further increase in σ is avoided because it may entail false negatives. Three iterations of this parameter search procedure are illustrated in Fig. 6.

Provided that the frame rate is sufficiently high, linear interpolation between two consecutive training points is a good choice for data representing the operation of a robot.

4 Results

This section provides a quantitative assessment of the proposed method and of the influence of various external conditions and deviations from the ideal case. The experiment is set up as follows. A training scene and a test scene is acquired showing a periodically working robot. During acquisition of the training scene, the robot operates normally. In the test scene we repeatedly violated the operation space of the robot by moving a hand in various ways inside that space (without collisions). Both movies are denoised by the adaptive filter described in [20]. Then, by applying PCA on the training scene, the classification subspace is selected. The OC-SVM classifier and the background detector are trained on the training scene as described above and applied to the test scene. In order to compare to a ground truth, we manually labeled all frames of the test scene based on visual inspection. The true positive rate is then the number of frames that are correctly classified to show operation space violations divided by the frames that actually are, according to the human inspector, such violations. In turn, the false positive rate is the fraction of frames classified as a violation while actually showing undisturbed operation. The results of all experiments are graphically presented in Figure 7. In all plots, the lines indicate the receiver operator characteristic (ROC) curve for varying discrimination threshold θ, whereas the circles indicate the accuracy achieved by the threshold that was automatically selected by the proposed heuristic.

As can be seen in the top left plot in Figure 7, the false positive rate of the OC-SVM classifier is very low as desired, whereas the true positive rate is not optimal. There are two reasons for this deviation. First, the heuristic used to train the classifier, optimizes on the true negative rate. The kernel size of the SVM is increased until all temporal interpolations of successive training samples are classified as valid machine
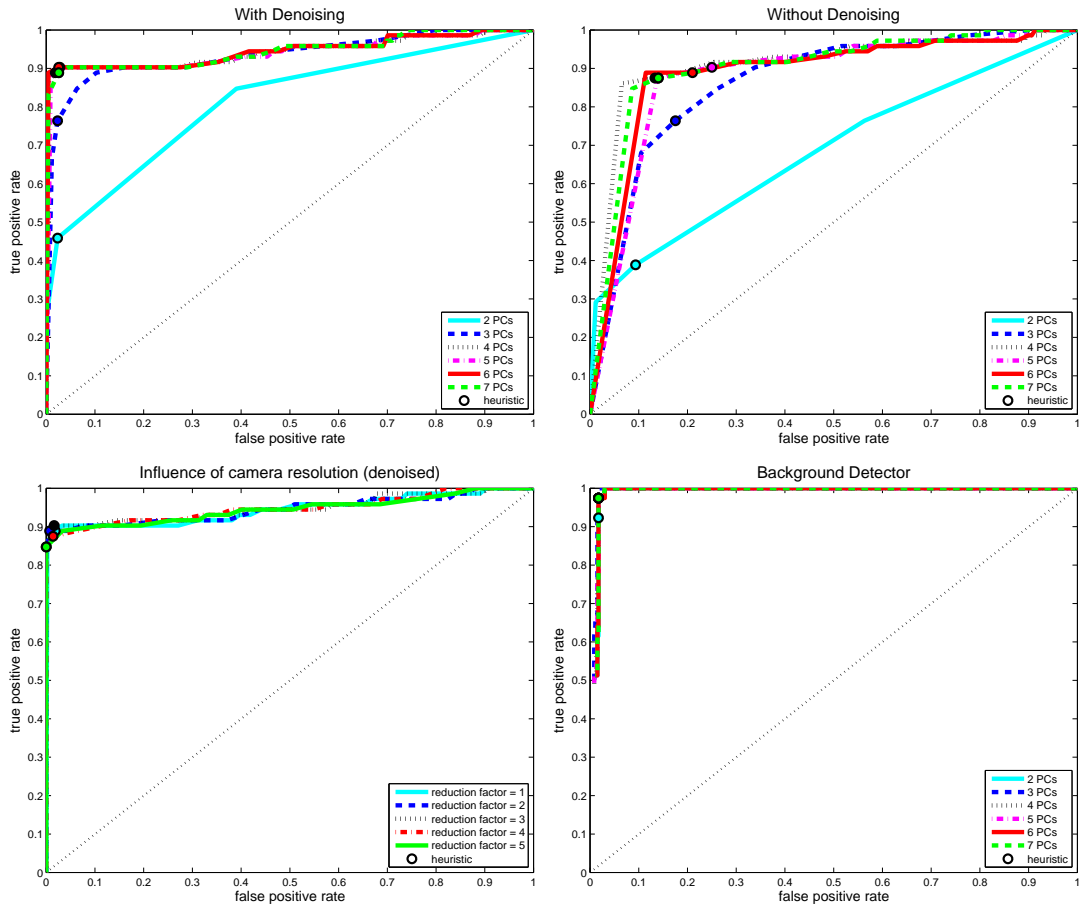
Fig. 7 Influence of the discrimination threshold on the classification error for various numbers of used principal components. The circles on the ROC curves highlight the threshold selected by our heuristic. Top left: default configuration with denoising of each frame. Top right: Results without denoising. Bottom left: Influence of camera resolution. Bottom right: Performance of the background mask; see discussion in the text.

states. Lacking positive training samples, the discrimination boundary can thus possibly come too close to the subspace of potential positive observations. These borderline cases contain either tiny intruders of the machine's operation space or bigger intruders that only slightly penetrate that space. Second, the manual labels given to the test set are perhaps overly conservative, in the sense that a careful human when inspecting a frame in which a hand almost reaches the maneuvering range, usually rates the frame as an operation space violation, even though this might not yet be the case.

Framewise inspection of the classification result supports both explanations. The frames with missed detections (positive manual labels and negative OC-SVM prediction) are always at the beginning or the end of the short sequences that show intrusions, as can be seen in Figure 8. These frames show the moment when the intruder is about to penetrate or leave the machine's operation space. Such borderline frames were always labeled as positive by the human expert, possibly including cases where there is no actual intrusion shown. In the video, given as supplementary material [1], the reader

may judge these cases his- or herself. In cases with actual intrusions, the part of the intruder that penetrates the operation space is so small that detection is very hard. However, in such cases, the complementary classification given by the background detector already raises an alarm which could be used for improved security.
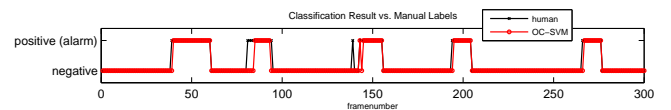

Fig. 8 Comparison of a classification result between OC-SVM and a human. False negatives of OC-SVM are predominantly at the beginning or the end of intrusion sequences.

4.1 Influence of Subspace Dimensionality

The top left plot in Figure 7 shows the ROC curve of six classifiers (and the curve for random guessing), each one trained

---

[1] www.mariofrank.net/monitoring/robot.avi

in a subspace of different dimensionality, reaching from two to seven. The performance is pretty robust to the number of principal components: choices between four and seven result in only negligible changes of accuracy. However, when using three or two principal components only, performance breaks down. The trend of the corresponding eigenvalues illustrated in Figure 3 already suggested that most variation is captured by the first four or five eigenvectors.

4.2 Influence of Noise

The experiment described above was carried out on denoised images [20]. Without filtering, the data exhibits strong motion artifacts and noise. In particular, those parts of the robot with very low infrared reflectivity exhibit high variance in the distance estimation. As can be seen in the top right plot in Figure 7, this significantly decreases the classification accuracy: the false positive rate increases. Therefore, and given that the denoising step can be performed fast, we advocate using a filter.

4.3 Influence of Camera Resolution

The resolution of the camera has only minor influence on the classification accuracy. In Figure 7 middle right, the ROC curves on sequences with decreasing image resolution are shown. The resolution of the original denoised images was reduced by a factor from 1 to 5 via averaging. For this particular kind of data, averaging gives more informative results than sub-sampling since it better mimics a design decision that can often be seen in the context of continuous-wave time-of-flight range cameras. The developers trade off resolution against precision of range estimation. Fewer but bigger image pixels result in a more reliable range estimation since the signal strength is integrated over a larger area (in fact, more recent implementations of this sensor type tend to have lower resolution for this very reason). When merging four pixels by averaging over them, we emulate a sensor with half the resolution along each dimension but equal chip size.

4.4 Accuracy of the Background Detector

The background segmentation method described in Section 3.1 provides a simple yet powerful classifier. The bottom right plot in Figure 7 depicts the ROC curves of this method against the dimensionality of the classification subspace. Already the first two principal components suffice to properly segment the background and thus to establish a reliable classifier. It should be stressed that the true positive rate in Figure 7 is overoptimistic due of the unbalanced test dataset available to us. Every frame with an obstacle in the robots operation space (a hand, see supplementary video) also contains an even bigger occlusion of the background (an arm). The false positives are still low because (i) the intruding hand traverses the background region fast, (ii) the double-variance threshold is very

insensitive to small background occlusions, and (iii) the human expert, as discussed earlier, tends to raise the alarm a few frames before the intruder actually penetrates the operation space. In summary, we advocate to use both classifiers together (as illustrated in the video provided as supplementary material). Up to a small sum and a threshold operation, the background mask is a byproduct of the dimensionality reduction step which makes its computational costs negligible.

4.5 Runtime

With the parameter σ set in advance, training with a MATLAB implementation took approximately three minutes on an Athlon64 at 2.2 GHz (2 GB RAM). Iterative parameter training (cf. Section 3.2) can raise this time up to 20 minutes. Note, however, that training has to be performed only once during setup, or when conditions or the operation mode of the robot change. In surveillance mode, the classification step itself is essentially instantaneous, while denoising is of the order of the frame rate.

5 Conclusion

We have presented an approach for monitoring a periodically moving machine. The main contribution of this article is a novel problem-specific regularization for the training algorithm that makes the configuration of such a monitoring system easy. After setting up the camera such that its field of view contains the machine plus a safety margin around it, the user simply has to acquire a few periods of unperturbed machine operation. Our method then parametrizes the classifier automatically. An important advantage of the method is that it works without detailed information of the geometry of the observed robot or its operation space. Moreover, there is no need to send signals from the robot controller to the classifier. It works completely autonomously and is therefore easy to set up.

The space behind the observed machine is occluded from the camera. Therefore, depending on the geometry, an additional detector with a complementary perspective may be required. A multi-detector setup can still be monitored with the algorithms introduced here: the measurements from the different detectors can simply be concatenated to obtain an augmented feature space; the rest of the calculations remains the same.

Since the algorithms involved do not directly rely on the fact that the images are depth images, they should, in principle, work with conventional gray-scale or color cameras as well. In practice, though, subtle changes in illumination will lead to false positives, we hence recommend to resort to a more illumination-invariant 3D camera.

The proposed system is conservative, in the sense that sudden sensor failure or occlusion or geometric perturbation will always lead to a departure from the training sequence, and hence to an alarm.

References

1. J. L. Novak, J. T. Feddema, "A Capacitance-Based Proximity Sensor for Whole Arm Obstacle Avoidance", *IEEE Proceedings of the International Conference on Robotics and Automation, pp.1307-11314* (1992)

2. J. T. Feddema, J. L. Novak, "Whole Arm Obstacle Avoidance for Teleoperated Robots", *IEEE Proceedings of the International Conference on Robotics and Automation, pp.3303-3309* (1994)

3. Y. Yu, K. Gupta, "Sensor Based Roadmaps for Motion-Planning for Articulated Robots in Unknown Environment: Some Experiments with an Eye-in-hand System", *IEEE Proceedings of the International Conference on Intelligent Robots and Systems, pp.1707-1714* (1999)

4. H. Noborio, Y. Nishino, "Image-based Path-Planning Algorithm on the Joint Space", *IEEE Proceedings of the International Conference on Robotics and Automation, pp.1180-1187, Seoul* (2001)

5. V. Lumelsky, E. Cheung, "Real-Time Collision Avoidance in Teleoperated Whole-Sensitive Robot Arm Manipulators", *IEEE Transactions on Systems, Man and Cybernetics, vol.23 no.1, pp.194–203* (1993)

6. E. Goujou, J. Miteran, O. Laligant, F. Truchetet, P. Gorria, "Human detection with video surveillance system", *Proceedings of the IEEE International Conference on Industrial Electronics, Control, and Instrumentation*, vol.2, pp.1179–1184 (1995)

7. K. Morioka, Joo-Ho Lee, H. Hashimoto, "Human-following mobile robot in a distributed intelligent sensor network", *IEEE Transactions on Industrial Electronics, vol.51, no.1, pp. 229–237* (2004)

8. P. Vadakkepat, P. Lim, L. C. De Silva, Liu Jing, Li Li Ling, "Multimodal Approach to Human-Face Detection and Tracking", *IEEE Transactions on Industrial Electronics, vol.55, no.3, pp.1385–1393* (2008)

9. G. Eckert, "Automatic Shape Reconstruction of Rigid 3D Objects from Multiple Calibrated Images", *Eusipco 2000 Proceedings, Tampere, Finland* (2000)

10. A. Meisel, "3D-Bildverarbeitung für feste und bewegte Kameras", *Vieweg Publishing, Fortschritte der Robotik Nr.21* (1994)

11. T. Gecks, D. Henrich, "Multi-Camera Collision Detection allowing for Object Occlusions", *37th International Symposium on Robotics (ISR 2006) / 4th German Conference on Robotics (Robotik 2006)* (2006)

12. D. Henrich, T. Gecks, "Multi-camera Collision Detection between Known and Unknown Objects", *2nd ACM/IEEE International Conference on Distributed Smart Cameras* (2008)

13. D. M. Ebert, D. Henrich, "Safe Human-robot-cooperation: Image-based Collision Detection for Industrial Robots", *IEEE International Conference on Intelligent Robots and Systems* (2002)

14. A. Kumar, "Computer-Vision-Based Fabric Defect Detection: A Survey", *IEEE Transactions on Industrial Electronics, vol.55, no.1, pp.348–363* (2008)

15. T. Sarmis, A. A. Argyros, M. I. A. Lourakis, K. Hatzopoulos, "Robust and efficient event detection for the monitoring of automated processes", *VIE 2006, Proceedings of the International Conference on Visual Information Engineering, pp.454–459* (2006)

16. M. Jäger, F. A. Hamprecht, "Principal Component Imagery for the Quality Monitoring of Dynamic Laser Welding Processes", *IEEE Transactions on Industrial Electronics, vol.56, no.4, pp.1307–1313* (2009)

17. R. Schwarte, H. G. Heinol, Z. Xu, K. Hartmann, "A new active 3D-Vision system based on RF-modulation interferometry of incoherent light", *Photonics East – Intelligent Systems and Advanced Manufacturing, Proceedings of the SPIE, vol.2588, Philadelphia* (1995)

18. T. Spirig, P. Seitz, O. Vietze, and F. Heitger, "The lock-in CCD - Twodimensional synchronous detection of light", *IEEE Journal of Quantum Electronics, vol.31, pp.1705–1708,* (1995)

19. M. Frank, M. Plaue, H. Rapp, U. Köthe, B. Jähne, F. A. Hamprecht, "Theoretical and Experimental Error Analysis of Continuous-Wave Time-Of-Flight Range Cameras", *Optical Engineering, vol.48, no.1* (2009)

20. M. Frank, M. Plaue, F. A. Hamprecht, "Denoising of Continuous-Wave Time-Of-Flight Depth Images using Confidence Measures" *Optical Engineering, vol.48, no.7* (2009)

21. S. Roweis, L. Saul, "Nonlinear dimensionality reduction by locally linear embedding", *Science, vol.290 no.5500, pp.2323–2326* (2000)

22. J. B. Tenenbaum, V. de Silva, J. C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction", *Science, vol.290 no.5500, pp.2319–2323* (2000)

23. B. Schölkopf, A. J. Smola, K. R. Müller, "Kernel Principal Component Analysis" *Lecture Notes In Computer Science, vol.1327, pp.583–588, Springer* (1997)

24. B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola: "Estimating the Support of a High-Dimensional Distribution" *Technical Report, Microsoft Research* (1999)

25. B. Schölkopf, A. J. Smola: "Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond", *MIT Press* (2002)

26. V. Vapnik, "The Nature of Statistical Learning Theory". *Springer Verlag, New York* (1995)

27. B. Schölkopf, A. J. Smola, R. C. Williamson, P. L. Bartlett: "New Support Vector Algorithms" *Neural Computation, vol.12, no.5, pp.1207–1245, MIT Press* (2000)

28. J. Weston, A. Elisseeff, G. Bakir, F. Sinz, "The Spider - Machine Learning Toolbox for Matlab", *http://www.kyb.tuebingen.mpg.de/bs/ people/spider/main.html, Dept. Empirical Inference, Max-Planck Institute for biological Cybernetics, Tübingen* (2006)

29. T. Zhou, D. Tao, X. Wu, "Manifold elastic net: a unified framework for sparse dimension reduction", *Data Mining and Knowledge Discovery, doi: 10.1007/s10618-010-0182-x* (2010)