Sputter Tracking for the Automatic Monitoring of Industrial Laser Welding Processes

M. Jäger, S. Humbert and F. A. Hamprecht

Abstract

The importance of laser welding in industry increases. Many welds have high quality demands and one possibility to satisfy the quality requirements is to monitor the welding process with high speed cameras. Laser welding is a highly dynamic process, it is therefore challenging to distinguish between normal process fluctuations and abnormal error events in the recorded sequences.

This paper investigates a novel classification method to automatically analyze the recorded welding sequences and robustly find the abnormal error events. To our knowledge it is the first time that a framework to detect and track sputters in welding sequences is proposed and evaluated. To achieve a high usability of the classification algorithm, in the training phase the user only needs to mark suspicious sequences, but does not need to label individual frames within the sequences. The framework is tested on two challenging datasets from real welding processes. The results show that the material particles can be tracked accurately. On a sample dataset, the new approach finds all erroneous welds with a small false positive rate and outperforms previously developed methods.

Index Terms

Automated visual inspection system, Laser welding, Particle tracking, Computer vision, Quality inspection

I. Introduction

In recent years, industries increasingly substitute conventional welding processes with laser welding units. The major benefits of lasers are that there is no mechanical contact with the work piece, that a high energy concentration can be achieved and that a high degree of automation is possible. However, laser welding is a highly dynamic and chaotic

process and thus vulnerable to process errors such as material sputters, weld break-ins or weld reinforcements. Although process errors occur rarely, it is vital to ensure that all faulty welds are detected, since errors can lead to a malfunction or a complete outage of the manufactured component part. Therefore quality sensitive welding processes have to be monitored. Several groups are working on algorithms to automatically analyze and control industrial welding processes using the outputs of various sensor types [1]–[12]. One possibility to automatically detect laser welding errors is to on-line monitor the laser welding process with a high-speed camera [2], [4]. When laser radiation interacts with the work piece, secondary radiation is generated. This radiation contains information about the process stability and can thus be used to detect process errors. A schematic setup to monitor the welding process is shown in Fig. 1.

In the remainder of this paper, we will focus on welding errors due to material sputters which can lead to pores in the welding seam. An example of this error event is shown in Fig. 2 in terms of four consecutive frames (64 $\times$ 64 pixels) of a recorded welding process. The big dark object in the middle of the frames is the melt pool (false color representation, dark pixels represent high intensities). A smaller object (a material particle) is flying away from the melt pool towards the lower right corner of the frame.

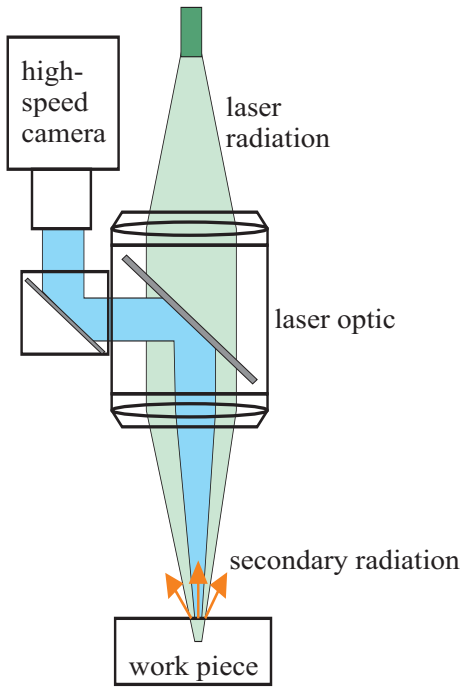

Fig. 1. Schematic setup for monitoring a laser welding process.

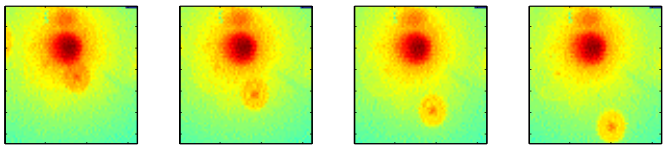

Fig. 2. In the middle of the frame the dark melt pool can be seen, from which a small material particle moves to the lower right corner. The field of view is approximately 3 mm $\times$ 3 mm. The cross, square and circle mark the predicted, measured and a posteriori position of the object according to the algorithm described in section II.

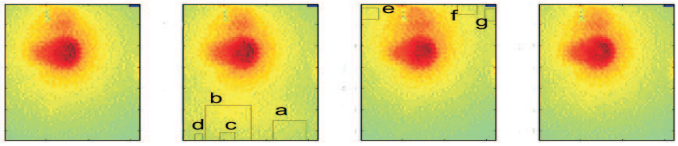

Fig. 3. Suspicious objects flagged by the change detection in consecutive frames. The brightness fluctuations found are not due to an error event.

During the welding process, the cylindrical part is rotated under the laser radiation and thus the position of the melt pool in the recorded frames does not change over time. As shown in Fig. 2, sputter events appear as dark objects in the recorded image sequences. Moving objects can be detected by analyzing the gray-level difference of successive frames [8], [9]. The change detection allows to find even weak brightness changes, but has the disadvantage that many false positives are found, as shown in Fig. 3. A high sensitivity of the change detection is necessary, since the material sputters can be small in size and low in intensity and for quality monitoring purposes it is necessary to avoid false negatives. It can be assumed that the sputter events are sufficiently long-lived to be observed in several consecutive frames of the recorded sequences.

In [10], a two-stage algorithm for process monitoring is introduced, the so-called *TISC*-method. It classifies suspicious events found in the change detection on a frame-per-frame basis and then aggregates these classification scores from consecutive frames. The algorithm does not establish whether two suspicious events from consecutive frames are related. Since random fluctuations of the brightness of background pixels can occur in consecutive frames due to the dynamic behavior of a welding process[1], a large number of false positives may result. In contrast, the approach proposed and tested in this paper uses a tracking algorithm to establish whether or not slightly suspicious objects that are flagged in different frames by the change detection algorithm may be related. Depending on this decision, the objects may

[1]The recorded melt pool of laser welding processes varies in its size and brightness. These brightness changes also effect the background pixels.

either be discarded as harmless fluctuations, or be considered as indicative of a welding error.

The paper is organized as follows: In section 2, the algorithmic framework is introduced. The performance of the tracking algorithm is validated using two different datasets in section 3. Conclusions are offered in section 4.

II. Tracking of Sputter Events

The Kalman Filter (e.g. [13]) is the most widely used algorithm for object tracking. It belongs to the family of Bayes filters, which estimate the state of a dynamic system from a sequence of noisy observations. The main advantage of the Kalman Filter is its ease of computation and memory efficiency. A disadvantage is that it is limited to linear systems with Gaussian noise and requires accurate sensors with high update rates [14]. If non-linearity and non-Gaussianity need to be included, possible extensions are the Extended Kalman Filter or the Particle Filter [15]. The Particle Filter is the most general tracking approach and a very flexible tool with a low implementation overhead, but higher computational cost. Since the tracking algorithm has to be deployed in a mass production environment with a rapid clock cycle, its computational efficiency is of great importance. With state of the art CMOS cameras it is possible to achieve a high update rate. The regular Kalman Filter with a linear system model and Gaussian noise has proven to be a suitable choice for the system under investigation [16].

A. System Description

A block diagram for the *TISC*-method is shown in Fig. 4(a) and for the tracking algorithm in Fig. 4(b). For both approaches, suspicious objects are flagged by the change detection algorithm. The *TISC*-method rates the detected suspicious objects in each frame using a polynomial classifier. The classifier outputs are then smoothed temporally and if the result exceeds a certain threshold, the sequence is marked as erroneous.

In the tracking approach, the change detection and feature extraction stages are followed by an object filter which rejects objects that are not relevant for the tracking procedure [2]. The object filtering is followed by the object tracking step (section II-B). The output of the object tracking are trajectories which are summarized with simple and rapidly computable

[2]The object filtering allows to reduce the computation time if a priori knowledge is available.

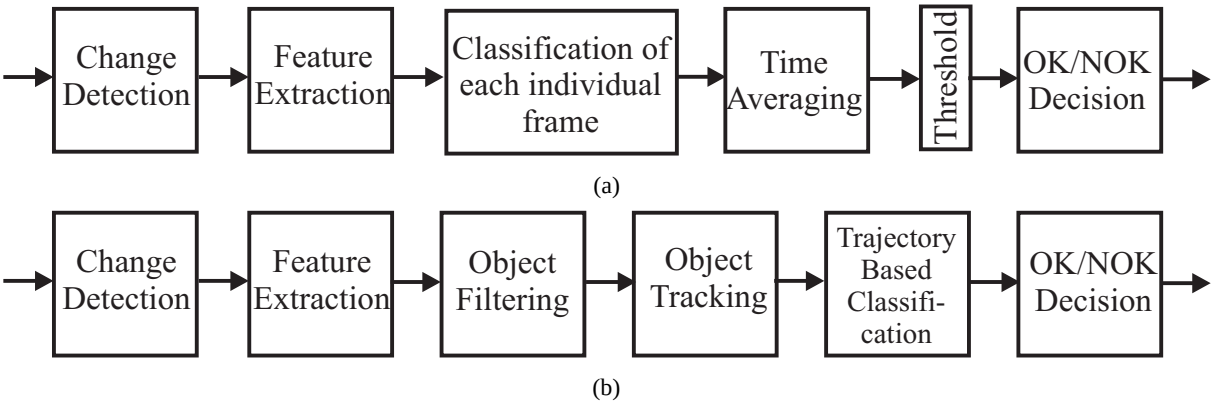

Fig. 4. Block diagrams for the *TISC*-approach (a) and object tracking and subsequent classification proposed here (b).

features such as direction and length of trajectory, or the average area of the tracked object. The decision whether or not a sputter really occurred is made in the trajectory feature space. Thus a single label is obtained for an entire welding process.

A disadvantage of the the *TISC*-approach is that it requires labels on a frame-per-frame basis. All suspicious objects in each frame have to be labeled manually if they belong to a sputter event during the training procedure. In contrast the tracking framework proposed here requires only a single label for an entire welding sequence. Labeled data is often scarce in practice. The tracking framework increases the computation time, but is an approach, which does not need object labels for training. In essence, these two approaches trade human for computational effort.

B. Object Tracking

The complete image sequence **S** consisting of T images I_t with $t \in 1, ..., T$ is available before starting the tracking procedure. The assumed system model **A** describes a constantly accelerated motion:

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 0.5 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0.5 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{1}$$

TABLE I
VARIABLES IN THE KALMAN ALGORITHM

$\mathbf{A}$	System model, describing a constantly accelerated motion
$\mathbf{y}_{t,i}$	Measurement vector of the i^{th} object in frame I_t
$\hat{\mathbf{x}}_{t,i}$	Prediction of the state vector using the system model $\mathbf{A}$ (see (3)) for the i^{th} object of frame I_{t-1} in frame I_t
$\mathbf{x}_{t,i}$	Estimated state vector of the i^{th} object in frame I_t after the measurement update
$\hat{\mathbf{P}}_{t,i}$	Covariance matrix of $\hat{\mathbf{x}}_{t,i}$
$\mathbf{P}_{t,i}$	Covariance matrix of $\mathbf{x}_{t,i}$

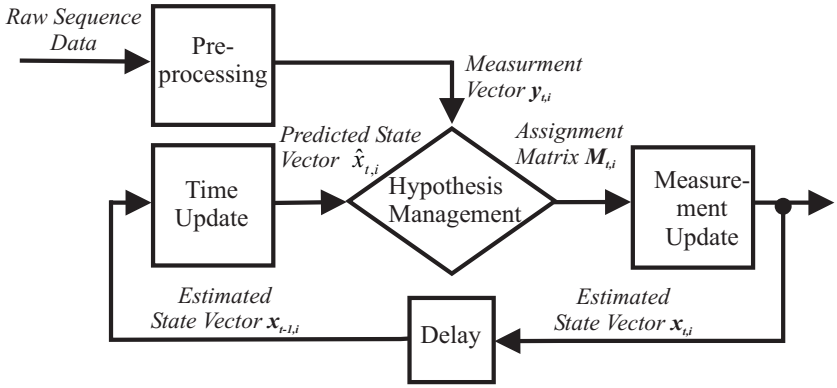

Fig. 5. Block diagram for the object tracking algorithm.

and the state vector $\mathbf{x}_{t,i}$, for the i^{th} object in frame I_t, is given by:

$$\mathbf{x}_{t,i} = (px, vx, ax, py, vy, ay)^T, \tag{2}$$

i.e. the coordinates for the center of mass, its velocity and acceleration in x and y-directions, respectively. Velocity and acceleration can only be estimated if the object is tracked at least over two and three consecutive frames, respectively. The system model $\mathbf{A}$ is used to predict the state vector $\hat{\mathbf{x}}_{t+1,i}$ for the i^{th} object in frame I_{t+1} from $\mathbf{x}_{t,i}$ with:

$$\hat{\mathbf{x}}_{t+1,i} = \mathbf{A}\mathbf{x}_{t,i}. \tag{3}$$

The notation for the variables used in the Kalman algorithm is summarized in Table I.

A block diagram of the tracking framework is shown in Fig. 5. In the *Preprocessing* stage, the change detection algorithm is applied to the entire recorded image sequence $\mathbf{S}$, outliers are marked in each frame and the positions of the segmented objects are determined. The

state vectors of the objects in the previous frame I_{t-1} are predicted for I_t in the *Time Update* step. It is possible that several objects are detected in a single frame, such that more than one sputter event has to be tracked. Since the Kalman Filter can only track a single object, multihypothesis tracking (MHT) is used to track several objects simultaneously [14]. MHT treats each hypothesis with a separate Kalman Filter. Their outcomes are combined in the *Hypothesis Management* step. MHT is computationally exponential in both memory and time [14]. Here however the number of simultaneously occurring sputter events is small (normally ≤ 3) and thus the computation time is not increased significantly. In addition, we assume that an observation can only be assigned to a single track and a track can only be the source of a single observation per frame. This reduces the complexity of the hypothesis management step. After the *Object Assignment* the velocity and acceleration of the sputters can be calculated and the predicted values can be reconciled with the observation using the Kalman gain. In the following, each step of the block diagram in Fig. 5 is explained in more detail:

1) Preprocessing: Following the change detection algorithm described in [8], each pixel $I_t(x, y)$ is normalized with:

$$I_{t,norm}(x, y) = \frac{|I_t(x, y) - med(I_t(x, y))|}{1.4826 \cdot med\,(\,|I_t(x, y) - med(I_t(x, y))|\,)}, \tag{4}$$

where x and y specify the pixel positions in each frame and med is the median operator in the temporal direction. Then, a binary image is obtained by thresholding:

$$I_{bin,t}(x, y) = \left\{ \begin{array}{ll} 0 & \text{for } I_{t,norm}(x, y) < bin \\ 1 & \text{for } I_{t,norm}(x, y) \geq bin \end{array} \right\} \tag{5}$$

where bin specifies the applied threshold. The normalization in (4) helps avoid the use of absolute intensities and thus makes the system more robust to variations in the weld materials, sensore degradation etc. The optimum was determined to be $bin = 3.9$ for sputter detection in laser welding sequences [8]. For the tracking approach, it is important to choose bin such that all sputter objects are flagged as suspicious objects. If bin is chosen too low, the processing time increases along with the number of suspicious objects.

2) Time Update: In this step, the state vectors for the objects found in I_{t-1} are predicted for I_t. Since it is possible that objects are lost due to the pre-processing step, a track

continuation function is implemented. The track continuation function predicts the state vector of trajectories in the absence of observations. The parameter $recall$ specifies the allowed number of consecutive frames in which it has not been possible to find an object for a given trajectory. If, after $recall$ consecutive frames, no further object for the trajectory is found, the track is terminated.

If an object appears for the first time, the predicted value $\hat{\mathbf{x}}_{t,i}$ is missing. Therefore $\mathbf{x}_{t,i}$ is initialized with the observed state vector $\mathbf{y}_{t,i}$ and the covariance matrix $\mathbf{P}_{t,i}$ is set to high values. The covariance matrices for the uncertainty of the system model and measurement model are kept constant during the tracking procedure.

3) Hypothesis Management: The trajectories describe the motion of a tracked object from the point at which it is detected until it is no longer observable. Objects belonging to the same trajectories are described with the assignment matrix $\mathbf{M}_{t,i}$, where $\mathbf{M}_{t,i}$ with $i = 1, ..., N_D(t)$ and $\mathbf{M}_{t,i} \in \{1, 2, ..., N_M\}$ specifies the index number of the trajectory to which the i^{th} detected object in frame t belongs to and N_M is the total number of found trajectories. For MHT, it is necessary to define an appropriate metric for the distance between the state vector for the currently observed and for the predicted object. In order to take the information of the covariance matrix into account the following probability assignment matrix is calculated:

$$P_M(\mathbf{y}_{t,i}, \hat{\mathbf{x}}_{t,j}) = \frac{1}{(2\pi)^d |\mathbf{P}_{t,i}|^{1/2}} \exp\left(-\frac{1}{2}\left(\mathbf{y}_{t,i} - \hat{\mathbf{x}}_{t,j}\right)^T \mathbf{P}_{t,j}^{-1}\left(\mathbf{y}_{t,i} - \hat{\mathbf{x}}_{t,j}\right)\right), \quad (6)$$

where i and j are varied over all measured and predicted objects in frame I_t, respectively. Only the values for the object positions are used for the calculation of (6). For the measured state vector, the velocity and acceleration are not yet known; thus $d = 2$ and $\mathbf{y}_{t,i} = (px, py)$. The observed objects are assigned to the trajectories in a bijective manner in order of decreasing P_M. Measured objects for which the maximum probability entries of P_M are below a certain threshold p_{th} are not assigned to an already existing trajectory and initialize a new trajectory.

4) Measurement Update: After matching the observed objects in I_t with the trajectories, the missing values for the velocity and acceleration in the measured state vectors can be calculated from the difference in the position and velocity of objects belonging to the same

trajectory in the previous frame. Now the predicted and measured values can be reconciled by using the Kalman gain to form the final estimate for the state vector $\mathbf{x}_{t,i}$.

III. Experiments

The tracking algorithm is evaluated on two different datasets recorded from real welding processes. *Dataset A* was recorded with a CMOS camera with 1000 frames per second (fps). In total, 129 erroneous sequences containing sputter events were obtained. The dataset is used to validate how accurately the Kalman filter can track sputter events. It was tested on challenging error events, where several particles had to be tracked simultaneously over a short period of time. In general, these events occur rarely.

The frame rate of the CMOS camera for *Dataset A* was relatively low. State of the art CMOS cameras offer a frame rate of up to 150 fps with a frame size of 1024×1024 pixels. Selection of a region-of-interest allows to trade spatial for temporal resolution; for instance, when restricting the region-of-interest to 40×40 pixels, up to 40000 fps can be achieved. In most welding applications a spatial resolution between 40×40 and 64×64 pixels is sufficient and the temporal resolution is then limited by the intensity of the back scattered radiations which prevents exposure times below a certain limit. Higher frame rates increase the temporal correlation and thus allow for better results with the object tracking algorithm. Therefore a second dataset - *Dataset B* - was investigated, which was obtained with a CMOS camera with a frame rate of 8000 fps. This dataset was compared with the procedure presented in [10] for quality monitoring of laser welding processes. In this application the sputters were caused by material remains from a preprocessing step; the material is burned upon interaction with the laser beam and this leads to errors on the weld seam.

A. Comparison between Manual and Automatic Object Tracking

Fig. 2 shows the results of the object tracking with the Kalman Filter for a sequence from *Dataset A*. For each frame, three different positions are visualized: the estimated object position (cross), the measured coordinates from the object segmentation (box) and the estimated state vector after the measurement update (circle). Since the particle is moving slowly and with constant velocity, the tracking algorithm can follow the object accurately.

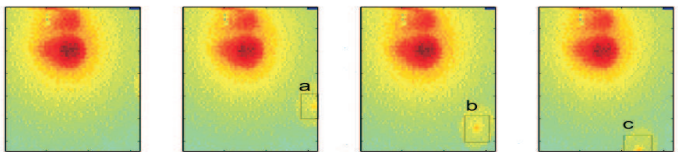

Fig. 6. Slowly moving particle. Given an appropriate probability threshold, the Kalman Filter tracks the object correctly.

TABLE II
ASSIGNMENT MATRIX FOR THE SEQUENCE SHOWN IN FIG. 6 AND FOR DIFFERENT PROBABILITY THRESHOLDS p_{th}. THE NUMBER IN THE COLUMNS BELOW A,B AND C SPECIFIES THE INDEX OF THE TRAJECTORY TO WHICH THE OBJECT IS ASSIGNED. THE MANUAL ASSIGNMENT IN THE LAST ROW IS CONSIDERED AS THE GROUND TRUTH. A LOW C-VALUE CORRESPONDS TO A HIGH AGREEMENT BETWEEN THE GROUND TRUTH (MANUAL) LABELS AND THE OUTCOME OF THE ALGORITHM.

	p_{th}	a	b	c	C
Kalman	10^{-8}	1	2	3	2
	10^{-16}	1	2	2	1
	10^{-20}	1	1	1	0
Manual	-	1	1	1	-

The first step in the evaluation was to compare the calculated trajectories of the proposed tracking procedure with the ones obtained by visual examination. The manual determination of object trajectories is regarded as ground truth in the following. In the 129 erroneous sequences of *Dataset A*, 2220 suspicious objects were found by means of the change detection algorithm (in a single sequence normally between $10-20$). Since the suspicious objects are found by temporal brightness changes, not all of the objects can be seen in the raw data (see e.g. Fig. 3). If the brightness of background pixels is relatively constant (small variance) at a certain position of the frame over time, small brightness changes are already sufficient to detect an imginary object.

In Table II the results for the manual and automatic determination of the assignment matrix for a sample sequence are shown. Visual inspection suggests that the objects in the three frames belong together. Hence, all objects have the same trajectory index number. The results show that the lower the probability threshold p_{th}, the better the tracking algorithm agrees with the manual trajectory assignment.

1) Performance Measure for the Tracking Algorithm: The C-value is a measure of how well the results from the algorithm correspond to the manual assignments. A low C-value corresponds to a high agreement between the ground truth labels and the outcome of the algorithm. The value of C is increased by one whenever an object is assigned to a different

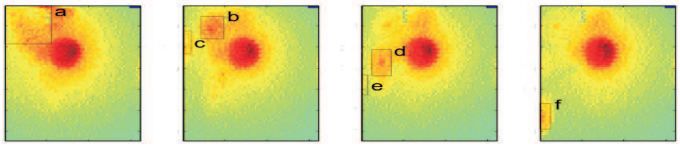

Fig. 7. Sputter event with ambiguous object assignment.

TABLE III
ASSIGNMENT MATRIX FOR THE SEQUENCE SHOWN IN FIG. 7

	p_{th}	a	b	c	d	e	f	C
Kalman	$1 \cdot 10^{-8}$	1	1	2	3	4	5	3
	$1 \cdot 10^{-16}$	1	1	2	2	3	4	3
	$1 \cdot 10^{-20}$	1	1	2	2	3	4	3
Manual	-	1	2	3	2	4	2	-

trajectory by the algorithm compared to the manual assignment. Therefore C has the value of 2 in Table II for $p_{th} = 10^{-8}$. Suspicious objects **b** and **c** are both incorrectly assigned to a new trajectory. If, after assigning the object to a wrong trajectory no further errors occur C is not increased. For instance, for $p_{th} = 10^{-16}$, the number of wrong assignments is $C = 1$. The object **b** is incorrectly considered as the beginning of a new trajectory, thus C is increased by one; but object **c** is then correctly assigned to the same trajectory as object **b** and C is not increased any further.

Fig. 7 shows an example of a challenging tracking scenario. Close to the melt pool irregularities are found which do not have sharp boundaries, as e.g. the objects in Fig. 6. This makes it difficult to obtain the ground truth labels. In such a case, manual assignments tend to be strongly subjective. Table III compares the assignments from the algorithm with the ground truth. The visual examination considers the objects **b**, **d** and **f** as a single particle that moves downwards very quickly. The tracking algorithm, bases its decision only upon the measured coordinates of the objects and groups objects which lie close together. In contrast, a visual examiner takes into account the shape and brightness of the moving particle. In addition, the tracking algorithm uses only the information from previous but not from upcoming frames. Since the complete sequence is available before starting the tracking algorithm, a Kalman Smoother could be used to overcome this shortcoming. A Kalman Smoother is a non-causal filter which bases the correspondence decision on the frames up to time index t and the upcoming frames (e.g. [17]).

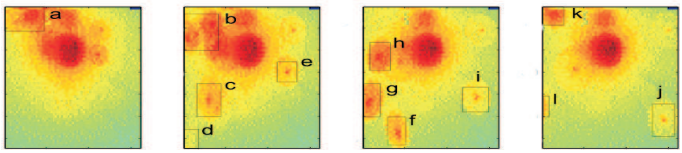

Fig. 8. Recorded image sequence of an explosion; many objects occure.

TABLE IV
ASSIGNMENT MATRIX FOR THE SEQUENCE SHOWN IN FIG. 8

	p_{th}	a	b	c	d	e	f	g	h	i	j	k	l	C
Kalman	10^{-8}	1	1	2	3	4	5	2	1	6	7	8	2	4
	10^{-16}	1	1	2	3	4	5	2	1	6	6	7	2	3
	10^{-20}	1	1	2	3	4	3	2	1	5	5	6	2	4
Manual	-	1	2	3	4	5	6	3	7	5	5	8	3	

Fig. 8 shows the recorded frames of an explosion. Many material particles are moving away from the melt pool. Table IV compares the assignments for this event. The visual examination assigns the objects **c**, **g**, **l** and **e**, **i**, **j** to two separate trajectories. The track **c**, **g**, **l** is also automatically detected by the algorithm for all used parameters. Objects **i** and **j** are assigned to one trajectory for p_{th} smaller than 10^{-16}. Object **e** is always regarded as the start of a new trajectory, since the distance to object **i** is too large.

Fig. 9 illustrates the effects of preprocessing errors on the tracking algorithm. The material particles are too close together, so that the threshold segmentation cannot detect them as separate objects. The two objects move from the melt pool to the lower right corner. In the first frame, they overlap completely and then travel with different velocities and thus separate in the following frames. The Kalman Filter is not capable of tracking the objects correctly due to the preprocessing errors. The tracking algorithm finds the trajectory **a**, **d**, **e** for $p_{th} = 10^{-20}$.

Fig. 10 evaluates the performance of the tracking algorithm for different probability thresholds between 10^{-16} and 10^{-3} for all of the 129 erroneous sequences. In general, the number of wrong matches decreases for lower probability thresholds. For a high probability threshold,

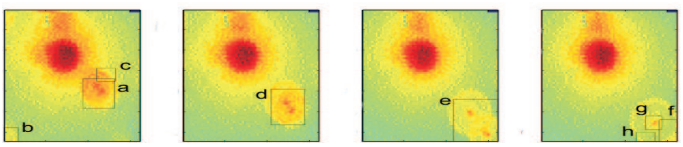

Fig. 9. Sequence with problems in the segmentation stage.

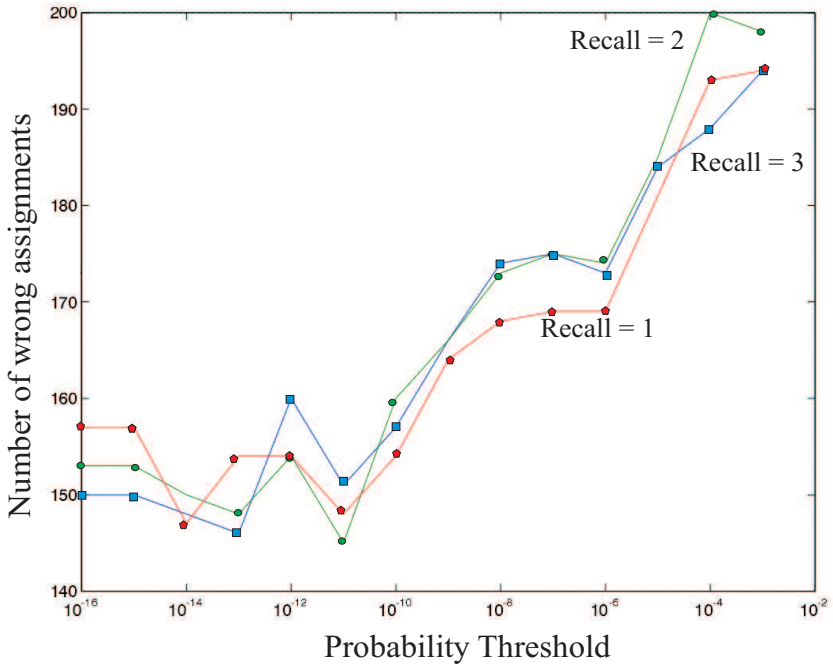

Fig. 10. Number of wrong assignments C of the tracking algorithm compared to the ground truth labels obtained from visual inspection, for dataset containing 2200 suspicious objects. 150 wrong assignments correspond to approximately 93% correct object assignments

the algorithm tends to assign objects to new trajectories even though they should belong to the same, whereas for smaller thresholds the algorithm groups the objects. From $p_{th} < 10^{-10}$, no further improvement can be obtained. The number of wrong assignments C varies between 145 and 160. This corresponds to approximately 93% correct object assignments. In addition, the influence of the $recall$ parameter is studied. It is varied between values 1 and 3. The graph shows that the influence of the $recall$ parameter on the performance of the tracking algorithm is negligible for *Dataset A*.

B. Object Tracking for Automatic Quality Control

Dataset B is obtained from a welding process with a larger dynamic range. So the number of detected suspicious objects is an order of magnitude higher (on average 0.1 suspicious objects are detected per frame) than for the *Dataset A* (on average 0.01 suspicious objects per frame). The parameters of the change detection are adjusted such that even weak sputter events can be found (bin = 3.7). The frame rate of the CMOS camera was 8000 fps and the duration of the welding process was approximately $0.4s$.

The test scenario consisted of 44 sputter sequences and 491 welding sequences without process errors. The error sequences were gathered from a real production process over a time

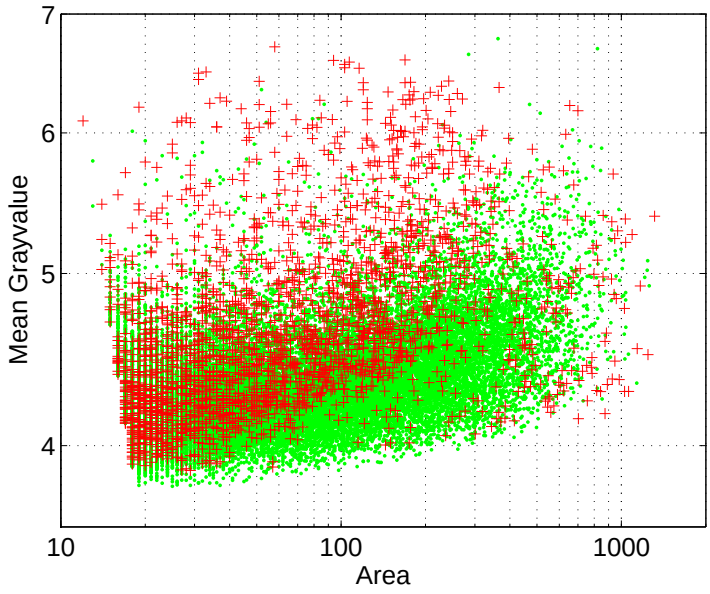

Fig. 11. Feature space of suspicious objects. As an example the mean gray value and the area of suspicious objects are presented. The false positives are marked with a circle, the sputter objects with a cross. The feature space shows a high degree of overlap between the two classes.

period of 3 months. In the recorded laser welding sequences, a large number of false positive objects are found. In Fig. 11, the feature space without using any sequence information is shown. The suspicious objects are manually labeled as false positives (circle) or sputter events (cross). Fig. 11 clearly shows the high overlap between the classes.

Both approaches sketched in Fig. 4 were compared. To ensure a fair comparison, the polynomial classifier was used both times. In both approaches, the feature selection was performed with a wrapper approach and the classifiers were trained using a 10 fold cross-validation so as to avoid false negatives (all parts with errors should be found).

In total, 13 object based features were calculated for the *TISC*-approach. The best feature combination of the first stage consisted simply of the mean gray value of a suspicious object, the polynomial degree was 1 and the optimal temporal filter length of the second stage was 2.

For the tracking approach, 11 trajectory features were calculated. The best feature combination consisted of 3 features (the aggregated track length and the start and stop position of the trajectory) and the optimal polynomial degree in the trajectory feature space was 3. The trajectory feature space of the tracking approach is shown in Fig. 12, which shows a clear separation between the normal and erroneous welding sequences.

TABLE V
SUMMARY OF THE RESULTS OF THE *TISC* APPROACH AND THE SPUTTER TRACKING AND SUBSEQUENT CLASSIFICATION. THE FALSE POSITIVE (FP) RATE IS DETERMINED SUCH THAT ALL ERRONEOUS SEQUENCES ARE FOUND.

	FP rate	Computation Times	Labelling Costs
TISC-approach	30.1%	≈ 5 s	High
Tracking & Classific.	2.7%	≈ 13 s	Low

The results show that the tracking approach can reduce the false positive (FP) rate from 30.1% for the *TISC* approach to 2.7% for the tracking and subsequent classification. The error rates were both determined with a 10-fold cross-validation for a false negative rate of 0%. The primary reason for the performance improvement is that due to the large number of false positive objects, the object based classification in the first stage of the *TISC*-approach shows poor results which cannot be compensated with the temporal filter in the second stage. Only the information on the correspondence of objects enables a separation. For comparison five sequences that showed only weak sputter events were eliminated from the dataset and the classification was performed again. In this case, the FP rate was 9.2% for the *TISC* approach and 1.2% for the tracking and subsequent classification. This shows that even when NOK borderline cases are eliminated, the tracking approach outperforms the *TISC* approach by approximately an order of magnitude.

Both approaches were implemented in Heurisko [18], a digital image processing software. The total processing time increased from 5s for the *TISC* approach to 13s for the object tracking on a Pentium 4 personal computer. Reasons for the large increase in the processing time are that the tracking approach could not be implemented without loops that have a high execution time in Heurisko, and that a large number of false positives are found. Since the clock cycle time was met by the implementation in Heurisko, no further improvements were considered. A further advantage of the tracking approach is that entire trajectories and not individual objects as in the *TISC* approach have to be labeled in the training stage. Therefore the labeling costs can be decreased dramatically. Table V summarizes the results of the two approaches.

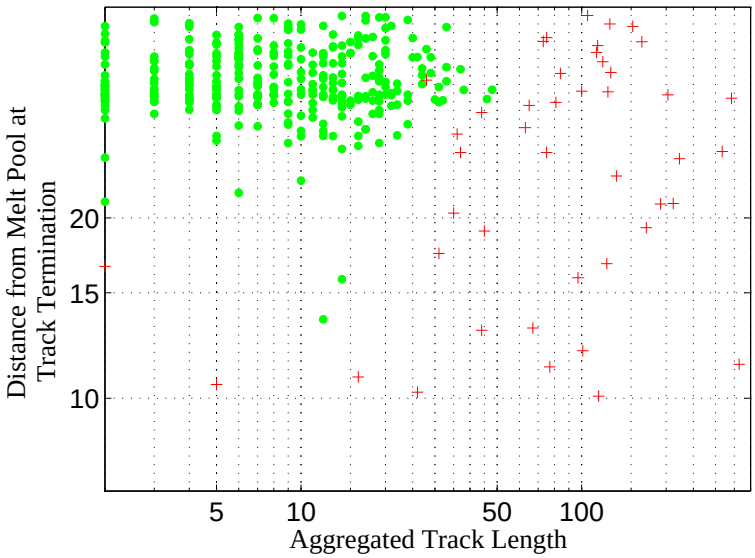

Fig. 12. Trajectory feature space with the erroneous sequences marked with crosses and the normal sequences with circles. The feature on the x - axis describes the aggregated track length (sum of the length of all detected trajectories within one sequence) and the feature on the y - axis shows the distance of the suspicious objects from the weld pool at the track termination. The overlap between the classes is reduced substantially in comparison to Fig. 11.

IV. Conclusions

We propose a tracking framework for the automatic online detection of sputter events in welding processes that can lead to critical pores in the weld seam. The training procedure avoids a tedious labeling of individual suspicious objects. We have compared the performance of the tracking algorithm with respect to ground truth labels obtained by visual examination of the sequences. With proper parameter settings, the tracking algorithm makes correct object assignments in up to 93% of all cases. We have also compared our method with the previously developed *TISC* [10] method for the quality monitoring of industrial processes. We were able to reduce the false positive rate by an order of magnitude to 2.7% for the given dataset.

An obvious limitation of the proposed approach is the representation of each suspicious object merely in terms of its position, velocity and acceleration, i.e. the loose coupling of the model with the underlying observations. Yet, experiments to include further object features such as the mean grayvalue or area in the system model only led to a slight improvement of the tracking performance. The main reason is that due to the change detection algorithm in the preprocessing stage, more detailed object features vary strongly from frame to frame for one object. Thus, no accurate system model for the Kalman filter could be found and a high uncertainty for the system covariance matrix had to be assumed.

REFERENCES

[1] C. Alippi, P. Braione, V. Piuri, and F. Scotti, "A methodological approach to multisensor classification for innovative laser material processing units," in *IEEE Instrumentation and Measurement Technology Conference*, May 2001.

[2] C. Kratzsch, "Realisierung eines kamerabasierten Prozessüberwachungssystems am Beispiel des Laserstrahlschweißens," Ph.D. dissertation, RWTH Aachen, 2002.

[3] J. Petereit, P. Abels, S. Kaierle, C. Kratzsch, and E. Kreutz, "Failure recognition and online process control in laser beam welding," in *Proceedings of* 21^{st} *International Congress on Applications of Lasers and Electro-Optics (ICALEO)*, 2002, pp. 91–98.

[4] J. Müller-Borhanian, C. Deininger, F. Dausiger, and H. Hügel, "Spatially resolved on-line monitoring during laser beam welding of steel and aluminum," in *Proceedings of* 23^{rd} *International Congress on Applications of Lasers and Electro-Optics (ICALEO)*, 2004.

[5] A. Bollig, S. Mann, R. Beck, and S. Kaierle, "Einsatz optischer Technologien zur Regelung von Laserstrahlschweissprozeßen," *Automatisierungstechnik*, vol. 53, pp. 513 – 521, October 2005.

[6] H. Junno, P. Laurinen, E. Haapalainen, L. Tovinen, and J. Röning, "Resistance spot welding process indentification using an extended knn method." Dubrovnik, Croatia: Proceedings of the IEEE International Symposium on Industrial Electronics, June 2005, pp. 7–12.

[7] D. Naso, B. Turchiano, and P. Pantaleo, "A fuzzy-logic based optical sensor for online weld defect-detection," *IEE Transactions on Industrial Informatics*, vol. 1, pp. 259–273, November 2005.

[8] M. Brocke, "Statistische Ereignisdetektion in Bildfolgen," Ph.D. dissertation, University of Heidelberg, 2002, http://www.ub.uni-heidelberg.de/archiv/3065.

[9] ——, "Statistical image sequence processing for temporal change detection," in *Pattern Recognition*, ser. Lecture Notes in Computer Science, L. V. Gool, Ed., no. 2449, 24th Annual Meeting of the German Association for Pattern Recognition. Zürich: Springer-Verlag, Heidelberg, 2002, pp. 215–223.

[10] S. Hader and F. A. Hamprecht, "Two-stage classification with automatic feature selection for an industrial application," in *Classification, the ubiquitous challenge*, B. Michaelis and G. Krell, Eds., Proceedings of German Classification Society 2004. Dortmund: Springer, 2004.

[11] L. Gong,C. Liu and X. F. Zha, "Model-Based Real-Time Dynamic Power Factor Measurement in AC Resistance Spot Welding With an Embedded ANN", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 3, pp. 1442–1448, June 2007.

[12] H. J. Koskimaki, P. Laurinen, E. Haapalainen, L. Tuovinen and J. Roning, "Application of the Extended $k$nn Method to Resistance Spot Welding Process Identification and the Benefits of Process Information", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 5, pp. 2823–2830, October 2007.

[13] D. E. Catlin, *Estimation, Control and the Discrete Kalman Filter*. Heidelberg: Springer-Verlag, 1989.

[14] D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello, "Bayesian filtering for location estimation," *IEEE Pervasive Computing*, vol. 2, pp. 24–33, July-September 2003.

[15] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, January 2001.

[16] S. Humbert, "Partikel-Verfolgung beim Laser-Schweißen mit dem Kalman-Filter," Master's thesis, University of Heidelberg, May 2005.

[17] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Upper Saddle River, NJ: Prentice Hall, 1999.

[18] *Heurisko, Digital Image Processing Software*, Hanau, Germany, http://www.heurisko.de.