

Received October 18, 2021, accepted November 12, 2021, date of publication November 16, 2021, date of current version November 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3128564

Robust Image Stitching and Reconstruction of Rolling Stocks Using a Novel Kalman Filter With a Multiple-Hypothesis Measurement Model

CARLO ALBERTO AVIZZANO¹, (Senior Member, IEEE), GABRIELE SCIVOLETTO²,
AND PAOLO TRIPICCHIO¹, (Senior Member, IEEE)

¹Institute of Mechanical Intelligence, Scuola Superiore Sant'Anna, 56125 Pisa, Italy

²Nextworks s.r.l., 56125 Pisa, Italy

Corresponding author: Paolo Tripicchio (paolo.tripicchio@santannapisa.it)

This work was supported in part by Trenitalia S.p.A.; and in part by the Italian Ministry through the Department of Excellence in Robotics & AI, Scuola Superiore Sant'Anna.

ABSTRACT This work introduces a novel algorithm for the reconstruction of rolling stocks from a sequence of images. The research aims at producing an accurate and wide image model that can be used as a Digital Twin (DT) for diagnosis, fault prediction, maintenance, and other monitoring operations. When observing large surfaces with nearly constant textures, metallic reflections, and repetitive patterns, motion estimation algorithms based on whole image error minimization and feature pairing with Random Sampling and Consensus (RANSAC) or Least Median of Squares (LMedS) fail to provide appropriate associations. To overcome such an issue, we propose a custom Kalman Filter (KF) modified by adding multiple input-noise sources represented as a Gaussian mixture distribution (GM), and specific algorithms to select appropriate data and variance to use for state prediction and correction. The proposed algorithm has been tested on images of train vessels, having a high number of windows, and large metallic paintings with constant or repetitive patterns. The approach here presented showed to be robust in the presence of high environmental disturbances and a reduced number of features. A large set of rolling stocks has been collected during a six months campaign. The set was employed to demonstrate the validity of the proposed algorithm by comparing the reconstructed twin versus known data. The system showed an overall accuracy in length estimation above 99%.

INDEX TERMS Kalman filter, Gaussian mixtures, image stitching, rolling stock, digital twin.

I. INTRODUCTION

Even if the national report on railway security [1], assesses Italy as one of the safest railways in Europe, in 2018 the average of significant railway accidents was one every 3.3Mln Tr-km (Millions of Train-kilometers), and the average number of deaths in train accidents was one over 5.133Mln Tr-km. When compared to the cumulative yearly train-mileage (384Mln Tr-km), this result shows about one-hundredth accidents per year. Major causes of train accidents are attributable to improper human behavior and (second) to maintenance issues.

Train maintenance is performed in two ways [2]: corrective maintenance and predictive maintenance. While corrective

The associate editor coordinating the review of this manuscript and approving it for publication was Farid Boussaid.

maintenance cannot be avoided, the allocation of economic resources in predictive maintenance can potentially reduce mechanical failure and then corrective maintenance.

In this work, we propose to reduce errors at the end of maintenance cycles and to detect when trains need to anticipate maintenance by setting up fault detection systems that exploit a reliable Digital Twin (DT). The goal is to move from the actual plan-based to condition-based maintenance that uses monitoring tools to assess the health status of the train.

DT concept was introduced in 2002 [3] as a method for Product Life-cycle Management [4]. However, the DT term was coined only in 2011 by NASA [5] as a conceptual basis in the astronautics and aerospace procedures. In modern maintenance paradigms, DT is considered as one of the most important tools in industry digitization [6]. Industry 4.0 approach,

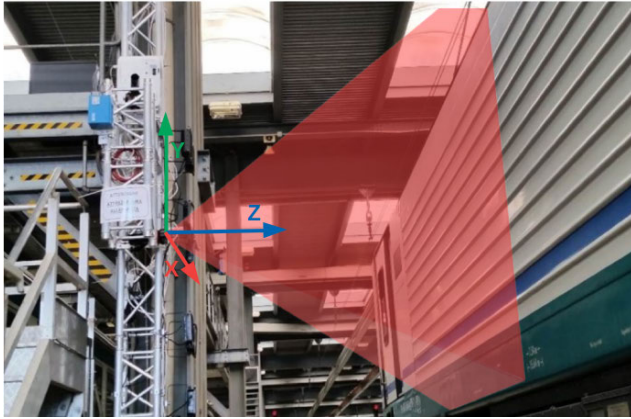


FIGURE 1. A concept of the wayside capture scenario using one camera and one laser.

in [7] imagines the predictive maintenance as the series of IoT networks that collect a large amount of data, followed by data fusion algorithms to produce a DT, and finally Artificial Intelligence techniques for taking decisions [8].

Boschert *et al.* [9], [10] introduced novel methods for the generation of railway-related twins both based on the identification of failures by comparing simulations on models and DT extracted from physical-data measurements. DT research commonly models train through laser/vision sensors: laser data are typically applied to wheels [11], carbon stripes [12], [13], pantographs and axles, while data from vision is being used in classical text detection as well as combined with deep learning [14]. Cha *et al.* [15] proposed a vision-based system to detect loosen bolts from images while Li *et al.* [16] provided a mechanism for serial number recognition from a fixed camera. Vision systems for rolling stocks analysis are applied to pantographs [17], [18] or to the entire train from different views [19], [20].

In this work, we aim to generate DTs for complete rolling-stock vessels for human or automated inspections [21]. This procedure will produce huge train bitmaps by composing multiple images through highly accurate mosaicing and stitching algorithms.

II. BACKGROUND

Extracting an accurate model of vehicle motion using vision information poses several problems such as data extraction, time-space alignment, sensor fusion, and uneven space-time measurements distributions [22]. Algorithms to analyze object motion in video or to align and combine images were explored in computer vision for almost three decades. An early example was the optical flow estimation, proposed by Lucas *et al.* [23]. Their approach detects object motion between frames by computing the relative spatial gradient. This method works on a local neighborhood of the moving object and successfully recovers the motion when an object moves slowly and shows a pattern that is not uniform and distinguishable from the background. The technique becomes

unstable in the presence of constant or repetitive patterns, rapid motions and/or background with similar patterns.

When an object occupies a large part of the scene, the same result may be achieved through image stitching. Image stitching (or mosaicing) works through an appropriate alignment and reprojection algorithm that could be estimated using two different approaches:

- **Direct Methods:** exploit a full hierarchical image analysis to evaluate pixel coherence and shift or warp the images consequently. These tools are used both for 2D and 3D reconstruction from camera images [24];
- **Feature-Based Methods:** use a subset of the image-pixels (features or keypoints) to evaluate the correspondence between frames.

Considering **Direct methods**, given two image frames (I_i, I_j), and a motion vector (s), two elements should be introduced. A transformation function that maps similar points between the two frames with coordinates (px_i, py_i) and (px_j, py_j) respectively:

$$(px_j, py_j) = T_f(px_i, py_i, s),$$

and an error-metric ($E(T_f, I_i, I_j, s)$) that computes how well the obtained transformation represents a good fit for the projection between frames. Therefore, the optimal fit can be found through a minimum search procedure:

$$\bar{s} = \arg \min_s E(T_f, I_i, I_j, s)$$

For instance, in case of pure translation along the x-axis, the motion vector reduces to scalar and simple metric functions may be employed like the *Sum of Squared Differences*:

$$E_{SSD}(s) = \sum_{px, py} [I_i(px + s, py) - I_j(px, py)]^2, \quad (1)$$

or the *Normalized Zero Mean Cross Correlation* [25]:

$$E_{ZNCC}(s) = \frac{1}{N} \sum_{px, py} \frac{1}{\sigma_n \sigma_{n-1}} [I_i(px + s, py) - \hat{I}_i] \times [I_j(px, py) - \hat{I}_j] \quad (2)$$

where i and j are the frame indices, σ represents the image pixel standard deviation, and \hat{I}_i, \hat{I}_j represent the average pixel intensity computed in the respective frames. These metrics will be employed in section IV-A to evaluate the performance of different methods. Concerning the simple case of a single translation, direct methods are usually computationally heavy and generally inaccurate when reflections and illumination properties, typical of real environments, alter the objects' appearance.

An alternative approach requires the extraction of a set of features from the images. Feature-points identify image coordinates that have very strong peculiarities, hence these methods show higher robustness to illumination properties. **Feature based Methods** are faster than Direct methods since they focus on a reduced number of matches. To cope with feature analysis errors during matching, usually, these methods

adopt outliers rejection tools such as RANSAC [26]. Some-time RANSAC is applied twice: first to identify keypoints belonging to the overlapped area, then to estimate the optimal homography matrix between frames. The idea of feature match was first exploited by Capel in [27] who proposed a way to achieve super-resolution images by mosaicing several images aligned and undistorted by a pre-computed homography map. The required homographies were estimated through a set of proposed point matches and a RANSAC procedure.

Algorithm 1 Frame Stitching and Retrieval of Inter-Frame Motion

```

procedure TrainStitch(frames[])
   $T_f \leftarrow$  motion model (e.g. translation)
  while  $I_i, I_j \leftarrow$  frames do
     $F_i \leftarrow$  FindFeature( $I_i$ )       $\triangleright$  Any Feature matcher
     $F_j \leftarrow$  FindFeature( $I_j$ )
     $s[i] \leftarrow$  RANSAC( $F_i, F_j, T_f$ )
     $Img \leftarrow$  JOIN( $Img, I_j, \text{OFFSET} = s[i]$ )
  end while
  return  $s[\cdot], Img$ 
end procedure

```

An example of the pipeline is shown in algorithm 1, first we run a feature detector algorithm on two consecutive frames, then we determine the optimal pixel shift ($s[\cdot]$) vector using a RANSAC estimator, finally we collate images together by translating the achieved super-resolution image by an offset of the estimated shift. The motion shift can also be converted to metric coordinate (s_x) if the camera matrix K [28] and the nominal train distance (d) are known:

$$s_x[i] = dK^{-1}s[i] \quad (3)$$

Brown & Lowe [29] refined such algorithm for mosaicing a panorama through a sparse set of images where the camera undergoes to pure rotations, and where all image points are far from the camera. Instead of using a full rigid transform, their approach simplified the homography ($H_{i,j}$) between frames as:

$$H_{i,j} = KR_i^jK^{-1} \quad (4)$$

where R_i^j represents the camera rotations between frames i, j . The homography was estimated matching sift-feature [30] with RANSAC. $H_{i,j}$ was furtherly refined using a tuning process known as *bundle adjustment* [31] that minimizes the points re-projection error in the target image coordinates. With this approach, Brown and Lowe simplified the Homography estimation problem (Eight-Degrees of Freedom) to only three parameters (relative rotations) and showed the efficacy of the matching search using only relevant features.

While Capel and Lowe approaches worked well on contrasted and opaque objects, constant patterns and metallic reflection diminish the number of valid features and create

several (reflection-based) false matches that should be eliminated. Industry 4.0 scenarios have plenty of plant situations where the ideal conditions to make these approaches working could not be fulfilled.

Pure motion reconstruction from images, does not handle constraints such as those provided by the target dynamics. Two common estimation tools help with this type of analysis: the Kalman Filter (KF) or the Particle Filter (PF). These filters combine the information from noisy measurements with an internal dynamic model and the prediction estimate from the previous model state. At each step, they predict an estimation of the model-state distribution, using a model that includes also a noisy input (uncertainty). The output, whenever available, may be used to correct this estimation. In the classical KF the inner model is linear and the noise/state models are represented by a Normal distributions [32], [33]:

$$\begin{cases} x_k = A_k x_{k-1} + B_k u_{m,k-1} + w_{k-1} \\ z_k = H_k x_k + v_k \end{cases} \quad (5)$$

In our case, we adopted a simple vehicle motion model, and we associated the filter state (x_k) with the estimated vehicle velocity (represented by s or s_x). Consequently, we have a unity process matrix $A_k = 1$, and $z_k = x_k$ is a direct observation of the state ($H_k = 1$). Since we have no information on the user input ($B_k = 0$), we assume that the input noise w_k includes also this information, while the output noise (v_k) models velocity measurement (as they could be estimated from feature matching). In the KF approach, both errors are uncorrelated, zero mean, and with known covariance respectively equal to Q_k and R_k .

In the PF instead, the stochastic process is simulated [34]. At each step, a large set of input-state combinations (namely particles) gets propagated through the system dynamics to identify how the population will evolve. The evolution is then refined whenever a measurement is available. Indeed PFs offer the possibility to model a wider set of input, noise, and state information, but even in modern implementations [35], [36], it comes with a heavy computational cost that is proportional to the number of particles.

During the past decades, there have been different efforts to extend the KF in different directions and particularly to cope with multivariate I/O data or systems with variable dynamics [37]–[39]. Ensemble Kalman filters are tradeoffs that make use of a reduced number of particle set distributions represented by Gaussians and handled as in KF.

To properly work, KFs assume the input and measurement data being affected by regular disturbances whose models can be reduced to a form of Normal distribution. However, in our case, the output signal derived from feature analysis is blended with reflections and background noise whose size and statistical properties profoundly alter the possibility of filtering out disturbances.

To overcome these weaknesses we decided to classify the input in clusters that represents the type of detected signal. For this purpose, we adopted a robust fit through a

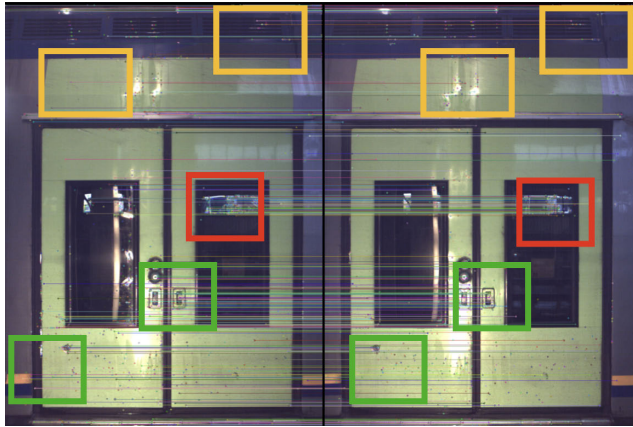


FIGURE 2. Two consecutive frames collected during the acquisition. Green boxes represent correct matches. Red boxes show wrong matches due to reflections. Yellow boxes highlight wrong matches due to the repetitive patterns.

Gaussian Mixture Model (GMM [40]). GMM is a common tool employed for data clusterization to find relevant components of multivariate distributions, i.e. a model where data are approximated by a weighted sum of Normal Distributions:

$$p(\theta) = \sum_{i=1}^N \phi_i \mathcal{N}(\mu_i, \Sigma_i) \quad (6)$$

where the probability density function ($p(\theta)$) is represented as the sum of a finite number (N) of Gaussians each with its own mean (μ_i), variance matrix (Σ_i) and cumulate probability (ϕ_i).

As we will show, the GMMs may act like a natural companion to KFs since each tool can feed the other with appropriate statistical information.

III. PROBLEM DEFINITION AND MODEL DEFINITION

This section presents the mathematical approach alongside a typical scenario that is exemplary of the data reconstruction problem. A set of cameras is placed wayside the railway to capture images of a train while it is passing. Capture framerate, illumination, exposure time, focal distance, and camera parameters have been calibrated to prevent distortion and blurring effects, to maintain an appropriate color range, and to ensure that the same details appear in a sequence of two or more frames. For the sake of simplicity, the cameras x-axes are assumed to be parallel to the navigation direction, while the z-axes are considered orthogonal to the principal train surface (see figure 1). Hence, the following assumptions can be made: 1) the motion model is a pure translation; 2) train features move on lines at a constant distance from the camera; 3) the presence of glasses and metallic paintings will generate a large number of reflections interpreted as fixed features (outliers); 4) some objects, such as doors, seats, internal illuminations, can generate a set of false linear motions. Figure 2 shows a typical scenario: only the green squares can be associated with effective train translations; red areas represent reflection effects and yellow boxes include

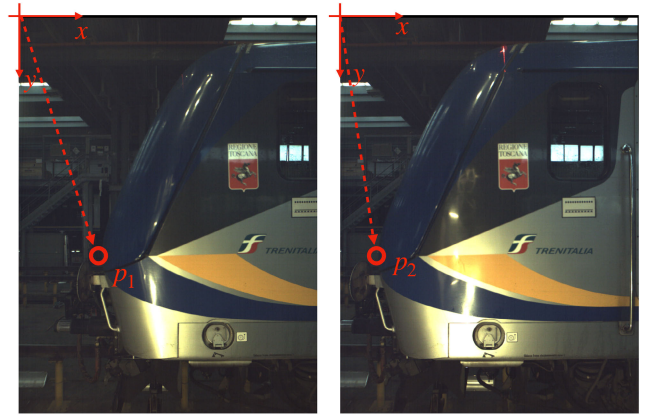


FIGURE 3. An example of the shift between consecutive frames.

features taken on different distance planes. In such a case, the use of typical feature matching methods to compute a homography transformation fails due to *missing data* [41], or excess of false data that do not represent the appropriate train motion.

A. IMAGE ACQUISITION AND CAMERA MODEL

The goal of the motion estimation is to provide a shift vector ($s(k)$) that describes the translation in pixels between frames. Since in our case the motion is considered to fall along the x-axis of the camera, we may assume that the train speed can be computed from the shift vector, the camera sensitivity (S_{xy}), the calibrated train distance d , and the camera framerate ($F = 1/T$) as:

$$v_x(t = kT) = \frac{dS_{xy}^{-1}s(kT)}{T} \quad (7)$$

A common approach is to find relevant points between frames (features) assuming a regular pre-calibrated perspective projection:

$$\begin{bmatrix} x/z \\ y/z \\ z \end{bmatrix} = P_a \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (8)$$

where $P_a = K_a [R_a; 0_{3 \times 1}]$ is the projection matrix, R_a represents the rotation of the camera with respect to an absolute reference frame (in our case the identity matrix) and K_a is the camera intrinsics matrix. Features can be detected using any of the common feature detection algorithms such as SIFT, BRISK [42], and ORB [43] to name a few. Once the features have been detected, a homography can be estimated by finding a correspondence between at least two matched-features (eight for a generic homography) as shown in figure 3 or using statistical tools that estimate the best homography-matching while rejecting outliers.

B. DATA AND DISTURBANCES

A density plot of motion histograms, useful for the assessment of the data distribution and for evaluating the importance of the reflections on the motion reconstruction,

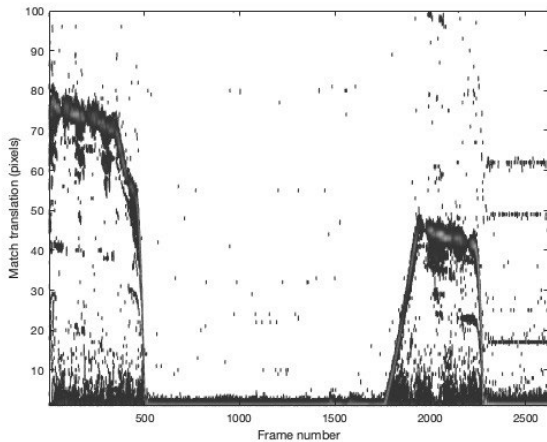


FIGURE 4. Histogram vs. frame. The figure shows the density of matched points for each camera-frames pair. The figure encodes, per frame, the profile of the train motion speed.

is reported in Figure 4. For each frame number (in the x-axis), the plot shows the density of all found matches. The Y-axis represents the detected motion shift in pixels-per-frame, which can be converted into train speed (m/s) through an appropriate conversion constant. The more the plot is intense, more features have been found indicating a similar train speed.

Figure 4 shows three typical issues: 1st the background noise generates several false matches, 2nd the reflections generate a high number of zero-translation matches that for certain frames can be higher than the number of effective matches, 3rd other illumination and motion artifacts can create regular distortions on the image patterns which maps into a considerable number of regular echoes on the histogram map. There are many situations in which the number of correct features is small and the most frequent value is not the target (expected) motion speed. This happens when the moving surface has large constant paintings and does not present enough recognizable features to match. Given these conditions, a motion detector that uses the only hypothesis of the prevalence of good matches cannot always estimate the proper motion. When the number of valid matches is small, a RANSAC procedure will fail to converge to the precise result with detrimental consequences on the overall stitching.

However, during the analysis, the *null-motion* hypothesis (zero translation) cannot be rejected, since, are not uncommon cases in which a train stops and restarts within the recording. Hence, the reconstruction algorithm should privilege the estimation of a relevant non-zero motion, without rejecting a priori the null-motion hypothesis as one of the possible solutions.

Looking more attentively to Figure 4 we may devise a strategy to select which would be the most appropriate estimation of the trajectory followed by the train by obeying a set of simple guidelines: 1. Random peak matches are provided by sensing disturbances; 2. the rolling stock is a physical object

and, as such, its motion should be constrained to a set of differential equations; 3. motion peaks non-coherent with the maximum train acceleration should be rejected; 4. a frame range with a coherent non-zero motion histogram sequence, will be associated to the train motion even if it is less likely than the null-motion hypothesis.

In particular, once we run GMM to cluster the motion histogram of Figure 8, we got a set of mean values (μ_i), representative weights (ϕ_i) and noise covariances (Σ_i) that are well suited to be used as Kalman candidates for the measurement (μ_i) and the noise covariance ($R_i \propto \Sigma_i$).

In this case a mixture of three different components is sufficient to represent: 1. the motion model; 2. the reflection model; and 3. all other noise measures. The GMM clusterization was performed through the Expectation-Maximization (EM) algorithm [40], which splits all motion pairs into J groups such that $N_k = \sum_{i=1}^J N_{i,k}$ is the total number of features matched in consecutive frames. Each group has its own size ($N_{i,k}$), average ($\mu_{i,k}$) and squared variance ($\sigma_{i,k}^2$), the latest estimated from group-samples.

However, sometimes EM may fail due to different random initializations or scarcity of valid features. A decision algorithm should, therefore, determine if one or none of the current predictions can be used for a correction step. Several “robustification” approaches for KFs are available (e.g. [39], [44]–[46]). However, these approaches are highly focused on the tail-shapes rather than the presence of multiple/strong (false) peaks in the observational model, so a newly dedicated approach is considered more appropriate. Our proposed model is based on the predicted-likelihood, thus requiring only one hyper-parameter, and more appropriately identifies the effects of disappearing relevant matches and rejects inappropriate measurements (false matches).

C. FILTER MODEL

The adopted filter model can be represented by a Kalman Filter employing the dynamical system represented in eq. 5 where the state $x_k = s_x(kT)$ embeds the velocity between frames as described in eq. 7, and it is characterized by a process covariance $P_k = \sigma_{x,k}^2$. The input-noise w_k is zero-mean and its variance encodes the maximum acceleration between frames provided by the driver on the train.

We also assumed the output of the model to be estimated as the most likely mixture average (μ_i) described in eq. 6, while the associated measurement variance (Σ_i) has been used to estimate the variance of the output related noise (v_k). Both μ_i and Σ_i were selected from the Gaussian Mixture using a specific algorithm (see Algorithm 2) that takes in input a shift population (δ_k) produced with feature pair-match among two consecutive frames, and eventually converted to metric distances through eq. 3.

The selection algorithm works as follows, first it determines which group is more coherent with the data predicted by the KF. This operation is performed by estimating the combined likelihood ($\mathcal{L}_{i,k}$) of each group mean ($\mu_{i,k}$), with

Algorithm 2 Selection of Kalman Best (KB) From a Motion-Shift Population δ_k

```

procedure  $\mathcal{S}_{KB}(\delta_k, P_{z_{max}})$ 
   $\{N_{i,k}, \mu_{i,k}, \sigma_{i,k}^2\} \leftarrow GMM(\delta_k)$ 
   $\mathcal{L}_{i,k} \leftarrow N_{i,k} \cdot Gauss(x_k - \mu_{i,k}, P_k) \triangleright$  KFLikelihood
   $i^+ = \max(\mathcal{L}_i) \triangleright$  SelectCondition
  if  $\text{erfc}(\text{abs}(\mu_{i^+,k} - x_k)/\sigma_{x,k}^2)) < P_{z_{max}}$  then  $\triangleright$ 
    RejectCondition
    return none
  else
    return  $z_k = \mu_{i^+,k}, R_k = \frac{\sigma_{i^+,k}^2}{(N_{i^+,k}-2)}$   $\triangleright$ 
    CandidateCorrection
  end if
end procedure

```

respect to the current Kalman prediction ($\mathcal{N}(x_k, P_k)$). Only the most relevant group will be taken into consideration, and new data estimation will be accepted only if the \mathcal{Z} score (computed using the Gaussian complementary error function, erfc) of the data candidate is significative (e.g. $z_{max} = 3 \rightarrow P_{z_{max}} > 99.85\%$). Hence we use the prediction variance (P_k) as a means to estimate the overall likelihood of a candidate observation. If new data are rejected (return value is none) the algorithm proceeds with the Kalman prediction, otherwise, it provides a candidate observation (z_k) and a candidate noise (R_k) for a Kalman correction step.

In the covariance estimation, we corrected the data covariance into a mean covariance using the number of matched features belonging to the group and keeping into account the effective degrees of freedom. The relevance threshold (z_{max}) is a parameter that should be selected experimentally. Choosing larger values will allow diverging shift-estimations to be accepted as correct. Setting it too low could make the filter discard proper measurements.

D. ROBUSTNESS ANALYSIS

To assess the capability of the system to adhere and maintain the proper tracking, two cases are considered: (1) complete lack of correct matches, and (2) a reduced number of good matches compared to the wrong ones. In what follows we will assume μ_G, σ_G, N_G as respectively being the mean, the variance, and the population of the mixture component associated to the good matches; μ_W, σ_W, N_W , the same data collected for the most likely wrong component in the GMM decomposition. Without loss of generality, we may consider only mismatches against the dominant component of GMM decomposition.

When no matches are available the algorithm will properly work with the KF prediction until the RejectCondition remains true. The number of correct prediction steps is therefore limited to:

$$k < \frac{(\mu_w - x_k)^2}{z_{max}^2 Q_0} \tag{9}$$

where we assumed that the input-noise process remains constant ($Q_k = Q_0$).

Application example

Considering a train capture with an average train arrival speed equal to $2m/s$, a capture framerate set to $20Hz$ and with smooth accelerations ($\approx 0.4m/s^2$) it is possible to compute $Q_0 \approx (.4/20)^2 = 4 * 10^{-4}$. The typical mismatch errors are in the order of the average train pixel-shift (for reflections) or 10% of it when some features belonging to different planes are matched (usually a quite small number). Computing k starting from this data, for instance using $Z_{max} = 2$, it is possible to obtain: $k_R = 2500$ frames (achieved in 125s) for robustness to reflections, and $k_{FP} = 25$ frames (1.25s) for features on false planes.

When only a small number of good features is available the KF-GMM algorithm will perform a good correction until the KFLikelihood condition returns good matches. A conservative limit threshold can be derived under the following theorem:

Theorem 1 (Distribution Switch): In steady state condition, with constant input, given a KF-GMM tracking as defined by eq. 5, with $\mathcal{D}_G = \{N_G, \sigma_G, \mu_G\}$ describing the distribution associated to the current correction-tracking, and $\mathcal{D}_W = \{N_W, \sigma_W, \mu_W\}$ an alternate tracking distribution isolated by GMM analysis, the filter will proceed correcting using \mathcal{D}_G if the following switch limit condition is respected:

$$\frac{N_G}{N_W} > e^{-\frac{(\mu_W - \mu_G)^2}{\sigma_G^2}}$$

Proof: First we consider the likelihood ratio logarithm to check when the \mathcal{D}_G is the most likely condition, i.e.: $\ln(\mathcal{L}_G/\mathcal{L}_W) > 1$, where each likelihood term may be expressed as:

$$\mathcal{L}_G = \frac{N_G}{\sum_i N_i} e^{-\frac{(x_k - \mu_G)^2}{\sigma_{x,k}^2}} \tag{10}$$

$$\mathcal{L}_W = \frac{N_W}{\sum_i N_i} e^{-\frac{(x_k - \mu_W)^2}{\sigma_{x,k}^2}} \tag{11}$$

That with few algebraic operation, lead us to:

$$\frac{\sigma_{x,k}^2}{2(\mu_G - \mu_W)} \ln \frac{N_G}{N_W} < \frac{\mu_G + \mu_W}{2} - x_k \tag{12}$$

If we introduce the hypothesis of steady state condition with constant input, we have:

$$\sigma_{x,k+1}^2 = \frac{\sigma_{x,k}^2 \cdot \sigma_G^2}{\sigma_{x,k}^2 + \sigma_G^2} < \sigma_G^2 \tag{13}$$

$$x_{k+1} \approx \mu_G \tag{14}$$

That reintroduced in eq. 12 lead us to the result. \square

Distribution switch example

In the case a wrong value is selected from the GMM correction, we may over-estimate the typical capture noise with $5 \cdot 10^{-1}m$, even in such case we got the following threshold percentages:

- Reflection mismatch ($\Delta\mu = 2m$): $N_G/N_W > e^{-\frac{2^2}{10^{-2}}} \approx 0$
- False plane mismatch ($\Delta\mu = .2m$): $N_G/N_W > e^{-\frac{.2^2}{10^{-2}}} \approx 0.018$

The presented KF-GMM algorithm allows to greatly improve tracking in the presence of great amounts of disturbances through rejecting false inputs that are typical of different robust fitting methods (e.g. RanSAC and LMedS). A practical example shows how even a small number of correct matches can dominate the tracking behavior and ensure that the filter will proceed with the correct estimation. Compared with particle filters, the data structure is much simpler to manage, and the computational load remains almost identical to that of the traditional KF.

IV. RESULTS COMPARISON

This section will present the results obtained from an experimental acquisition campaign. A comparison with existing methods is performed on data captured on a maintenance facility in Osmannoro (Firenze, IT), during a period of 2 months, which presents almost two trains capture per day. Comparing stitching results obtained from different approaches requires particular care. When the largest percentage of matches corresponds to good features, whichever algorithm is used, it leads to the same good result. However, when rebuilding a complete DT, even a few misaligned frames lead to an inappropriate reconstruction of the whole train. Hence, to show the limitations of alternative algorithms, an analysis has been applied on long real frame-sequences (> 1500 frames) like those shown in figure 4.

A. DIRECT METHODS

To evaluate direct methods the error metrics defined in equations 1 and 2 have been used. The motion shift-vector (u) was computed by minimizing the error metric while restricting the search in the interval $u \in [0, 150]$ (pixel per frames).

Figure 5 shows an example of using eq. 1 in real-case scenario. The figure shows the value of the metrics computed in four different frames distant less than 20 frames (1s) each other. While in the first frame (#167) could be argued that the train is moving at 78 pixels per frame, as soon as more reflections (caused by a window) appear in the scene, a new global minimum appears (#180, #197), and the previous minimum, now local, tends to disappear (#207).

Estimating the train moving speed only using the absolute, or relative minimum policy leads to reconstruct only a portion of the correct motion shift. These results are summarized in

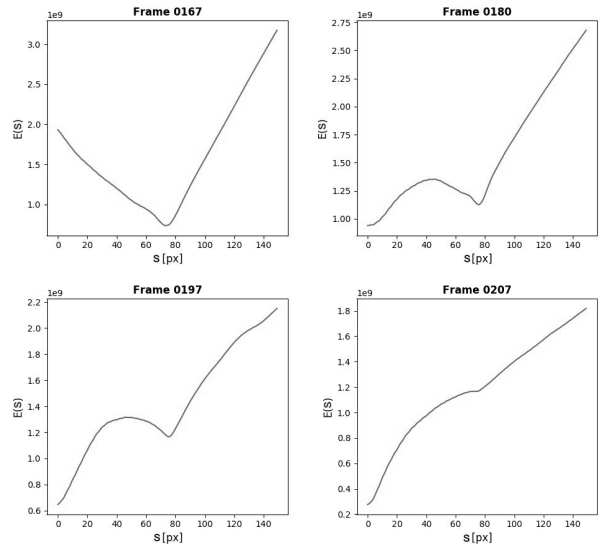


FIGURE 5. E_{SSD} performance on four different frame twins ($n = 167, n = 180, n = 197, n = 207$). X coordinate represents the motion shift (s) in pixels and Y value represent the index value ($E(s)$).

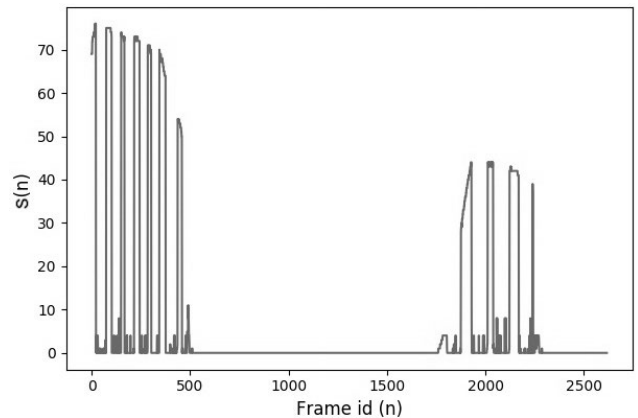


FIGURE 6. Motion shift vector $s(n)$ computed through the E_{SSD} minimization algorithm.

figure 6 where the estimated motion-shift (ESSD) is plotted over time (frame number).

It is possible to observe that whenever the image pattern is constant, under-exposed, over-exposed, or repetitive, the reflections dominate over the details alignment and many minimum motion-vectors are equally probable.

In this case, the reflections (see figure 2) generate several false null-motion estimates for about one half of the whole capture sequence. Moreover, while for some frames (#180, #197) this problem could be avoided by searching for a local minimum different from zero, this minimum may completely disappear in subsequent frames.

A similar result was determined when using the E_{ZNCC} scoring function. The table 1 (discussed later) summarizes the success rate in the reconstruction achieved by both direct and feature-based methods.

As a result of one, or a few subsequent wrong motion shifts the whole carriage reconstruction becomes distorted.



FIGURE 7. A critical example of carriage reconstruction using E_{SSD} (left) vs. proposed algorithm (right).

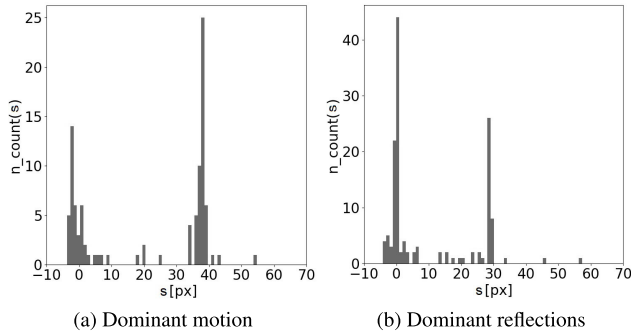


FIGURE 8. Histograms of matching pairs using SIFT+Horizontal constraint. X-axis represents the motion vector, while Y-axis shows the number of pairs found.

Figure 7 shows the effects of a critical reconstruction achieved by applying *Direct Methods* on a real train.

B. FEATURE-BASED METHOD

One of the limitations of the Direct Methods lays in the fact that all image pixels are considered to be equals unregarding the fact they belong to a dark or saturated area or have a uniform texture. We know instead that when a constant image is moved in front of a camera, the information should be extracted only by a few relevant points that are clearly distinguishable from the background. Feature-based methods help in this regard as they greatly reduce the number of points to be used for estimating the motion, and introduce complex descriptors that facilitate the matches between frames.

The additional use of the RANSAC tool helps to eliminate incoherent motion shifts that are not confirmed by a large percentage of the matches. However, even introducing these two tools, the distinction between the artifacts of figure 2 cannot be performed, being both coherent and feasible solutions to the problem.

Accuracy outcomes achieved with different stitching algorithms while reconstructing some critical train parts such as windows, doors, and the whole train have been computed. Table 1 compares these results between direct and indirect methods. While the percentage of mismatched motion shift is greatly diminished, it still remains a consistent percentage of the overall number of frames.

None of the tested algorithms succeeded in reconstructing a complete train image or even a single carriage since the number of reflection outliers was so high that it frequently happened they overcome the number of correctly matched

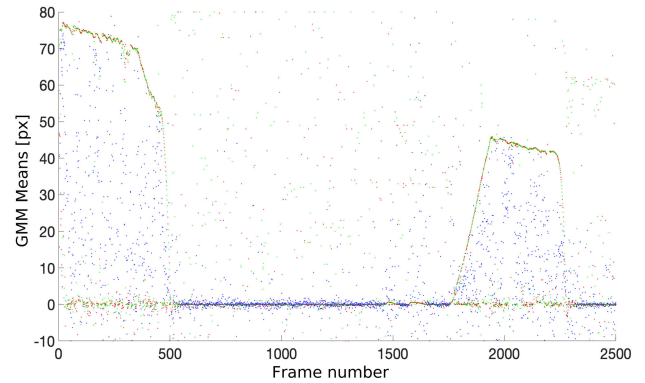


FIGURE 9. The figure shows the velocity groups estimated through GMM clustering.

features. The introduction of the RANSAC procedure slightly improves the results but it is still limited to a small fraction (about 20%) of the elements on the carriage surface.

A simple sample and hold (SH) strategy can be introduced to improve the reconstruction quality: maintain the last known speed, when newer acquisitions are incoherent with the previous ones ($abs(u_k - u_{k-1}) > threshold$). This procedure greatly improves the quality of the reconstruction but still leaves open two issues:

- maintaining the same speed, when the train is accelerating, introduces length reconstruction errors on some train details;
- especially at low speeds, it is particularly critical to determine which thresholds to apply to discriminate incoherent motions.

C. PROPOSED SOLUTION

To explain why the proposed algorithm outperforms traditional ones, it could be useful to analyze the results of the GMM algorithm shown in Figure 9. The graph displays for each frame the three average values of the GMM decomposition (μ_i) in different colors (green, red, and blue) ordered

TABLE 1. Performance comparison between direct and feature-based methods.

Method	SSD	ZNCC	SIFT	SIFT + RANSAC
Failed motion detection	68.9%	69.1%	21.9%	20.5%
Component Len. Error	65.1%	67.3%	23.4%	21.3%
Train Failure Rate	100%	100%	100%	100%
Window Failure Rate	100%	100%	95.5%	81.2%
Door Failure Rate	100%	100%	94.2%	77.1%

TABLE 2. Numerical performance of the algorithm.

Element Type	Element Count	Real Width [px]	Avg Meas. Width [px]	Std Dev	Err%
Door A	146	1215	1215	1.75	0
Door B	136	888	883	4.87	0.56
Window A	566	840	833	5.59	0.83
Window B	961	850	842	7.72	0.94

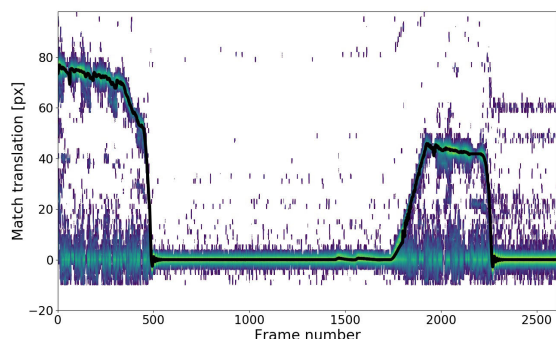


FIGURE 10. The figure shows the estimated velocity reconstructed by the algorithm in overlay to the contour plot of the captured features.

by relevance (ϕ_i). The alternation of red and green dots in the graph illustrates that the population size criterium is not adequate to discriminate between motion and reflections.

Here the two improvements from Algorithm 2 come to play. First, we re-score the relevance into a likelihood that also depends on previous Kalman statistics; secondly, we ignore any estimations that significantly deviate from the prediction. While the first rule properly reorders measurements coherently with the current Kalman estimation, the second rule intervenes to reject any measurements in case there were no relevant features to estimate the motion.

When a measurement is considered valid, its extracted information (mean, STD, and population size) is fed to the Kalman correction step. In such a way the process noise description is continuously adapted to the current capturing conditions. In figure 10, a contour plot of the motion-sample density is overlaid with a black reconstructed line which derives from the proposed multi hypothesis Kalman filter.

The reconstructed trajectory has been found stable in all the captured scenarios and the algorithm capable to reject the disturbances coming both from the reflection features and from the object/features moving on different planes. Due to the Kalman “memory”, the algorithm follows properly the train acceleration-deceleration phases even when motion features were missing.

Surprisingly the algorithm detects properly even when the train motion restarts. This could be motivated by the fact that the train stopped in a particularly favorable situation, having a consistent number of proper features detected.

The accuracy of the proposed algorithm was validated on about 250 carriages, related to 40 different train captures, each composed by 1500/3000 frames, depending on the speed profile and length of the rolling stocks. Two types of

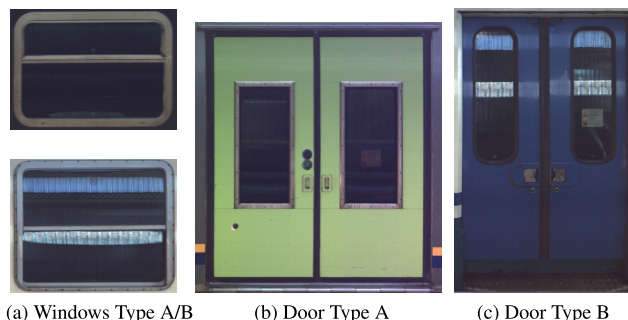







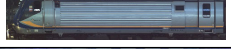



FIGURE 11. Four types of elements rebuilt during train motion.

numerical evaluations were performed: by focusing on door and windows details, or by estimating the whole carriage length.

The first analysis benefits from the fact that the elements may also be captured by single picture frames, and the result compared with the stitching sequence made of tenths of frame-strips. The elements rebuilt with this type of analysis are shown in figure 11. We focused on four element types: two doors and two windows. In this analysis, we can use the single picture as ground truth to estimate the percentage error committed by the stitching algorithm. The knowledge of the exact dimension of the element is not required. Table 2 shows the results of the analysis. The table contains the name of the element taken from the Figure 11, the number of element analyzed, the width of the element measured on the single frames, and in the three last columns the mean, the standard deviation in pixels, and the relative error (in percentage) of the estimated width as detected after the stitching algorithm. The measurements were taken manually from the resulting pictures and might be affected by $\pm 1px$ inaccuracy. In all cases, we got a correct reconstruction (100% success) and the error between the measured size in pixel and the average detected size was below 1% with an std lower than 8 pixels (1%) in the worst case (window type B).

When analyzing the carriages and the whole trains, since there was no effective ground truth to compare the reconstructed image with, the lengths of different carriages have been compared using the obtained standard deviation as an indicator of the algorithm robustness. The achieved results are shown in Table 3. For each carriage typology, the table presents the carriage type as an image, the number of carriages of the same type, the estimated length, and the standard deviation both in pixels and as a percentage in relation to the whole length. Even in this case, the algorithm showed excellent capabilities, even better than those related to individual

TABLE 3. Numerical comparison of the estimated carriage length.

Carriage Type	Carriage Count	Width [px]	Std Dev	Std Dev%
	63	16960	34.5	0.22%
	17	16966	20.3	0.11%
	14	10047	28.3	0.28%
	5	16962	13.9	0.08%
	4	17296	10.2	0.05%
	10	9911	15.8	0.15%
	63	16739	32.8	0.19%
	4	17151	12.2	0.07%
	6	17170	18.5	0.1%

elements. We explained this behavior due to averaging and quantization of errors between frames.

V. CONCLUSION

We presented a novel algorithm that combines two relevant tools for robust estimation: Gaussian Mixture Models and Kalman Filters. These tools were applied in a tracking application with the aim of stitching images of a train taken by a fixed camera. We highlighted how the two tools can be combined together offering at once a light and robust tracking system that is much lighter than a particle filter but still allows coping with data mixed with different noise types. In particular, GMMs provide time-variant statistical measurement models whose mean and covariance can be used in the correction phase, and KFs provide methods to discriminate or reject GMM's analysis in a highly disturbed environment.

Using a selection algorithm in the data filter model, allowed us to reject outliers when proposed sensor measures are incoherent or weakly related to the projected model. Additionally, we derived an estimation of robustness in two limit conditions and showed that it was solid enough for vision tracking tools.

The resulting filter was applied to a stitching case-study in a real field environment, under different lighting conditions and in presence of several measurement disturbances. The results showed the high reliability of the proposed approach with 100% of trains fully reconstructed without evident errors and with accuracy on estimated geometries greater than 99%.

The algorithm is currently employed by the national train company to rebuild train information using a single array of cameras that observe a train during its passage. Once rebuilt the whole image of the train, the facility also provides to detect particular elements on the train (serial numbers, windows, boxes, grids, etc.) and check their integrity by comparing them with original images taken just after a maintenance procedure.

ACKNOWLEDGMENT

The authors are grateful to the funding entities and to all the maintenance staff in Trenitalia who gently assisted their work.

REFERENCES

- [1] *Rapporto Annuale Sulla Sicurezza Delle Ferrovie. Techreport 2018*, Agenzia Nazionale Sicurezza Ferroviaria, Florence, Italy, 2018. [Online]. Available: <https://www.ansf.gov.it/rapporti>
- [2] Y.-H. Cheng and H.-L. Tsao, "Rolling stock maintenance strategy selection, spares parts' estimation, and replacements' interval calculation," *Int. J. Prod. Econ.*, vol. 128, no. 1, pp. 404–412, Nov. 2010.
- [3] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems*. Cham, Switzerland: Springer, 2017, pp. 85–113.
- [4] V. Gecevska, P. Chiabert, Z. Anisic, F. Lombardi, and F. Cus, "Product lifecycle management through innovative and competitive business environment," *J. Ind. Eng. Manage.*, vol. 3, no. 2, pp. 323–336, Oct. 2010.
- [5] E. Glaessgen and D. Stargel, "The digital twin paradigm for future NASA and U.S. Air Force vehicles," in *Proc. 53rd AIAA/ASME/ASCE/AHS/ASC Struct., Struct. Dyn. Mater. Conf. 20th AIAA/ASME/AHS Adapt. Struct. Conf. 14th AIAA*, 2012, p. 1818.
- [6] W. Kritzing, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018.
- [7] R. Rosen, G. Von Wichert, G. Lo, and K. D. Bettenhausen, "About the importance of autonomy and digital twins for the future of manufacturing," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 567–572, 2015.
- [8] P. Tripicchio, G. Camacho-Gonzalez, and S. D'Avella, "Welding defect detection: Coping with artifacts in the production line," *Int. J. Adv. Manuf. Technol.*, vol. 111, nos. 5–6, pp. 1659–1669, Nov. 2020.
- [9] S. Boschert, C. Heinrich, and R. Rosen, "Next generation digital twin," in *Proc. TMCE*, I. Horvath, J. P. S. Rivero, and P. M. H. Castellano, Eds., Las Palmas de Gran Canaria, Spain, 2018.
- [10] S. Kaewunruen and N. Xu, "Digital twin for sustainability evaluation of railway station buildings," *Frontiers Built Environ.*, vol. 4, p. 77, Dec. 2018.
- [11] A. Cavuto, M. Martarelli, G. Pandarese, G. M. Revel, and E. P. Tomasini, "Train wheel diagnostics by laser ultrasonics," *Measurement*, vol. 80, pp. 99–107, Feb. 2016.
- [12] E. Di Stefano, E. Ruffaldi, and C. A. Avizzano, "Automatic 2D-3D vision based assessment of the attitude of a train pantograph," in *Proc. IEEE Int. Smart Cities Conf. (ISC)*, Sep. 2016, pp. 1–5.

- [13] S. Judek and L. Jarzebowicz, "Algorithm for automatic wear estimation of railway contact strips based on 3D scanning results," in *Proc. Int. Conf. Expo. Electr. Power Eng. (EPE)*, Oct. 2014, pp. 724–729.
- [14] P. Tripicchio and S. D'Avella, "Is deep learning ready to satisfy industry needs?" *Proc. Manuf.*, vol. 51, pp. 1192–1199, Jan. 2020.
- [15] Y.-C. Cha, K. You, and W. Choi, "Vision-based detection of loosened bolts using the Hough transform and support vector machines," *Automat. Construct.*, vol. 71, pp. 181–188, Nov. 2016.
- [16] B. Li, B. Tian, Y. Li, G. Xiong, and F. Zhu, "A vision-based serial number recognition algorithm for HSR trains by nearest neighbor chains of connected components," in *Proc. IEEE Int. Conf. Veh. Electron. Saf.*, Jul. 2013, pp. 36–41.
- [17] P. Capece, L. Chiesi, A. Pinazzi, T. Kelsey, C. Wilson, M. Sabbatelli, M. Sacchi, and L. Venturi, "PANTOBOT-3D: An automatic vision-based inspection system for locomotive pantographs," in *Proc. IET Railway Condition Monit.*, 2016, pp. 1–5.
- [18] G. Karaduman, M. Karakose, and E. Akin, "Condition monitoring platform in railways based on IoT," in *Proc. Int. Conf. Artif. Intell. Data Process. (IDAP)*, Sep. 2018, pp. 1–4.
- [19] G. Lisanti, S. Karaman, D. Pezzatini, and A. D. Bimbo, "A multi-camera image processing and visualization system for train safety assessment," *Multimedia Tools Appl.*, vol. 77, no. 2, pp. 1583–1604, Jan. 2018.
- [20] G. Scivoletto, C. A. Avizzano, and P. Tripicchio, "A novel stitching algorithm for reconstructing rolling stocks from fixed cameras," in *Proc. 1st I-RIM Conf. Intell. Robot. Mach.*, Rome, Italy, 2019, pp. 157–158.
- [21] R. K. W. Vithanage, C. S. Harrison, and A. K. M. DeSilva, "Importance and applications of robotic and autonomous systems (RAS) in railway maintenance sector: A review," *Computers*, vol. 8, no. 3, p. 56, Jul. 2019.
- [22] Y. Wang, Y. Liu, H. Fujimoto, and Y. Hori, "Vision-based lateral state estimation for integrated control of automated vehicles considering multirate and unevenly delayed measurements," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2619–2627, Dec. 2018.
- [23] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Darpa Image Understand. Workshop*, Vancouver, BC, Canada, 1981, pp. 121–130.
- [24] H. Kim and B. Lee, "Robust 3-D object reconstruction based on camera clustering with geodesic distance," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 2, pp. 889–891, Apr. 2019.
- [25] L. Di Stefano, S. Mattocchia, and F. Tombari, "ZNCC-based template matching using bounded partial correlation," *Pattern Recognit. Lett.*, vol. 26, no. 14, pp. 2129–2134, Oct. 2005.
- [26] M. A. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [27] D. Capel and A. Zisserman, "Automated mosaicing with super-resolution zoom," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 1998, pp. 885–891.
- [28] R. Szeliski, "Image alignment and stitching: A tutorial," *Found. Trends Comput. Graph. Vis.*, vol. 2, no. 1, pp. 1–104, 2007.
- [29] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *Int. J. Comput. Vis.*, vol. 74, no. 1, pp. 59–73, Aug. 2007.
- [30] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 1150–1157.
- [31] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Proc. Int. Workshop Vis. Algorithms*. Berlin, Germany: Springer, 1999, pp. 298–372.
- [32] G. Welch and G. Welch, "An introduction to the Kalman filter," *Proc. SIGGRAPH, Course*, vol. 8, nos. 27599–23175, p. 41, 2001.
- [33] H. Winter, V. Willert, and J. Adamy, "Increasing accuracy in train localization exploiting track-geometry constraints," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 1572–1579.
- [34] S. Thrun, "Particle filters in robotics," in *Proc. 18th Conf. Uncertainty Artif. Intell.* San Mateo, CA, USA: Morgan Kaufmann, 2002, pp. 511–518.
- [35] S. Li, S. Zhao, B. Cheng, E. Zhao, and J. Chen, "Lightweight particle filter for robust visual tracking," *IEEE Access*, vol. 6, pp. 32310–32320, 2018.
- [36] P. Tripicchio, M. Unetti, N. Giordani, C. A. Avizzano, and M. Satler, "A lightweight SLAM algorithm for indoor autonomous navigation," in *Proc. Australas. Conf. Robot. Automat. (ACRA)*, 2014, pp. 2–4.
- [37] B. L. Boada, M. J. L. Boada, and H. Zhang, "Sensor fusion based on a dual Kalman filter for estimation of road irregularities and vehicle mass under static and dynamic conditions," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 3, pp. 1075–1086, Jun. 2019.
- [38] R. Li, V. Prasad, and B. Huang, "Gaussian mixture model-based ensemble Kalman filtering for state and parameter estimation for a PMMA process," *Processes*, vol. 4, no. 2, p. 9, Mar. 2016.
- [39] S. Kar, B. Sinopoli, and J. M. F. Moura, "Kalman filtering with intermittent observations: Weak convergence to a stationary distribution," *IEEE Trans. Autom. Control*, vol. 57, no. 2, pp. 405–420, Feb. 2012.
- [40] M.-S. Yang, C.-Y. Lai, and C.-Y. Lin, "A robust EM clustering algorithm for Gaussian mixture models," *Pattern Recognit.*, vol. 45, no. 11, pp. 3950–3961, 2012.
- [41] A. Ozerov, M. Lagrange, and E. Vincent, "GMM-based classification from noisy features," in *Proc. Int. Workshop Mach. Listening Multisource Environ. (CHIEME)*, 2011.
- [42] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2548–2555.
- [43] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.
- [44] R. J. Meinhold and N. D. Singpurwalla, "Robustification of Kalman filter models," *J. Amer. Statist. Assoc.*, vol. 84, no. 406, pp. 479–486, Jun. 1989.
- [45] K. L. Lange, R. J. A. Little, and J. M. G. Taylor, "Robust statistical modeling using the t distribution," *J. Amer. Stat. Assoc.*, vol. 84, no. 408, pp. 881–896, 1989.
- [46] Y. Huang, Y. Zhang, Z. Wu, N. Li, and J. A. Chambers, "A novel robust student's-t-based Kalman filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 3, pp. 1545–1554, Jun. 2017.



CARLO ALBERTO AVIZZANO (Senior Member, IEEE) received the Ph.D. degree. He is an Associate Professor of mechatronics and a Coordinator at the PERCRO Laboratory. He is an author of about 200 papers in the areas of mechatronics, control, computer vision, and robotics. His research interests include robotics, haptics, and intelligent systems for HRI, Industry 4.0, medicine, and transportation.



GABRIELE SCIVOLETTO is a Research and Development Software Engineer at Nextworks s.r.l. and formerly at Scuola Superiore Sant'Anna. His primary research interests include the Internet of Things, computer vision, and data analysis.



PAOLO TRIPICCHIO (Senior Member, IEEE) received the master's degree in automation engineering from the University of Pisa and the Ph.D. degree in perceptual robotics from the Sant'Anna School of Advanced Studies. In 2012 and 2016, he was a Lecturer with the Instituto Tecnológico de Orizaba, Mexico. He is currently an Assistant Professor with the Department of Excellence in Robotics & AI, Institute of Mechanical Intelligence, Scuola Superiore Sant'Anna. He is also currently the Scientific Program Manager at the Gustavo Stefanini Advanced Robotics Research Center, La Spezia, Italy. His main research interests include human-robot interaction, field robotics, computer vision, and AI.