Comparison of deep learning-based methods in multimodal anomaly detection: A case study in human–robot collaboration

Lin Yang¹, Wu Yan² and Hongmin Wu²

¹Guangdong Provincial Key Laboratory of Electronic Information Products Reliability Technology, Guangzhou, China
²Guangdong Key Laboratory of Modern Control Technology, Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangzhou, China

Abstract

Human-Robot Collaboration (HRC) has been widely used in daily life and industry for maximizing the advantages of humans and robots, respectively. However, the internal modeling errors or external perturbations still affect robotic systems such as human collisions and environmental changes. Multimodal anomaly detection plays an increasingly important role in HRC applications, which detects unexpected anomalies from multimodal signals. Due to the complex temporal dependence and stochasticity, it is still difficult to choose a common model applicable to all collaborative tasks, and lack of comparative analysis of existing methods and verification of specific application cases. In this paper, six representative deep learning-based methods are evaluated and the comparing metrics including detection accuracy, multi-modality combinations, and anomaly time bias. For a fair comparison, each detector models multimodal signals from non-anomalous samples and then determines an anomaly using a predefined threshold. We evaluate the detectors with force, torque, velocity, tactile, and kinematic sensing during a human-robot kitting experiment that consists of six individual skills, results indicate that the LSTM-DAGMM based detector outperformed the others, which yielding higher accuracy and efficiency. The metrics are measured with the RUC and ROC by changing the settings of multi-modality combinations and various anomaly biases, which aim to obtain the best performance of multimodal anomaly detection.

Keywords

Deep learning, multimodal anomaly detection, human robot collaboration, performance comparison

Corresponding author:

Hongmin Wu, Guangdong Key Laboratory of Modern Control Technology, Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, No. 100 Xianliezhong Road, Guangzhou 510070, China.
Email: hm.wu@giim.ac.cn
Introduction

As the rapid development of collaborative robot, HRC tasks (e.g. household, services, warehouse logistics, etc.) are willing to be accomplished in a shared, unstructured, and nonstandard environment.\textsuperscript{1–3} It's intractable to completely model the underlying dynamics, which likely to result in robotic abnormal movements with unexpected anomalies such as human collision, tool collision, and object slip in a HRC kitting experiment,\textsuperscript{4,5} as shown in Figure 1. Robots that can detect and respond appropriately to common anomalies have the potential to provide more effective and safer collaboration. Once the anomaly precisely detected, the anomaly identification for identifying the type, location, size of anomalies such that they can be used to prevent potential failures and minimize damage as well as recover the common anomalies.\textsuperscript{6}

An anomaly detector should detect when the current multimodal observations differ significantly from those past normal observations, which can be analogous to the problem of finding unexpected patterns in multivariate time series. The anomaly detection of multivariate time series using data-driven method remains a big challenge because of the complex temporal dependence and stochasticity during robot manipulation tasks.\textsuperscript{7,8} Several common factors that make the detection difficult in HRC contexts, including

1. The diversity of manipulation task, users, and environments makes the normal observations with noises;
2. The anomalies in robotics are usually occur from various sources such as body damage, sensing errors, control failures, or environmental changes;
3. The anomalies exist unexpectedly and occur sporadically. Lacking of realistic anomalous samples for training, the detection method has to be an unsupervised one;
4. The anomalies are multimodal that integrated multiple sensory signals from different sensors;
5. The range of anomalous observations are ambiguous, which would be determined by a point or a short sequence. The collection of anomalies is various;

Figure 1. Illustration of those most likely unexpected anomalies in a human-robot collaborative kitting experiment (left-to-right).
HC: human collision; OS: object slip; TC: tool collision; WC: wall collision.
The importance of modality combination for anomaly detection at different execution phase;

The fixed threshold and fast detection may increase the false positive rate (detect the normal as anomaly).

Specifically, for the reasons (1), the robot manipulation tasks are designed to adapt the modifications of human’s behaviors or environmental changes in HRC, which possibly resulting the robotic movements are modeled with the movement primitive techniques, for example, Dynamical Movement Primitives (DMP),\textsuperscript{9} Gaussian Mixture Model (GMM)\textsuperscript{10} as well as Kernelized Movement Primitives (KMP).\textsuperscript{11} Various complex manipulation tasks would be generated with the learned movement primitives by sequencing. For the reasons (4) and (6), although the robots can precisely understand the manipulation object and the interacting environment with the benefits of multimodal observation, how to evaluate the integrated sensors are redundant or insufficient given specific task remains challenging. Besides that, As described in Aryal et al.,\textsuperscript{12} the anomaly detector should address the problem of the multimodal data that represented using different units/scales. Therefore, the performance comparison of anomaly detectors under various multimodalities combinations is proposed in this paper, which can be used to determine the importance of each modality and more specifically the relationship of other modality values. This research can also directly understand how much each modality affects the manipulation task at different execution phase. Those aforementioned phenomenon emphasize the difficulties of effectively implementing anomaly detection in HRC scenarios.

To address these concerns, a wide variety of data-driven multimodal anomaly detector are popularly proposed in recent years.\textsuperscript{13–15} Motivated by the continued success of deep learning methods in high-dimensional data modeling, an increasing number of deep-learning methods have been applied in anomaly detection, the most advantage that do not require the effort of significant data preprocessing and feature engineering. In this paper, we mainly compare the performance of reconstruction-based methods that using autoencoder paradigm,\textsuperscript{16} where the anomaly thresholds are calculated based on the reconstruction error. The main idea of those considered methods in this compassion is to learn robust latent representations to capture normal dynamics of multimodal observations, which including both the temporal dependence as stochasticity. Therefore, although fruitful progress has been made in recent years,\textsuperscript{17} designing a robust anomaly detector on multimodal or high-dimensional sensory data in unsupervised fashion remains an open issues.

**Related work**

Typically collaborative robots used in HRC are equipped with multiple sensors that allow robots to perceive the internal information and external environment. The use of multiple sensors results in the generation of a sensory multimodal signal that
can be exploited to learn a robot’s model of the world. Therefore, robot abnormal behaviors are classified as such when at least one signal dimension exhibits inconsistencies compared to previously experienced nominal executions.

Generally, anomalies can be classified into four categories: point anomalies, collective anomalies, contextual anomalies, and change points. These categories have a significant influence on the type of anomaly detection methods that is employed. Specifically, anomalies in temporal data are contextual anomalies by time providing the context, where an abnormal observation is viewed in a particular context not just reflect the current moment. Another critical consideration is the dimensionality of temporal data. If multidimensional dataset is provided and each feature is represented in a time-series individually, one can use anomaly detection methods for multivariate time series. However, alternative methods that separate the temporal data for implementing anomaly detection in multidimensional space by univariate time series, respectively.

The recent advancement of deep learning-based anomaly detection methods have gained much popularity with their promising performance. For instance, implementing the anomaly detection using the Auto-Encoder (AE) by inspecting its reconstruction errors. Additionally, Recurrent Neural Networks (RNNs) architectures and have resulted in outstanding performance for a variety of problems including time series prediction and sequence-to-sequence learning. Until recently, many applications of deep learning in modeling temporal data involving univariate or multivariate for anomaly detection are presented good performance. Specifically, RNNs represent a significant improvement in efficiently processing and prioritizing historical information valuable for future prediction. When compared to dense Deep Neural Networks (DNNs) and early RNNs, the LSTMs have been shown to improve the ability to maintain memory of long-term dependencies due to the introduction of a weighted self-loop conditioned on context that allows them to forget past information in addition to accumulating it. LSTMs have been widely used in modeling time series in applications as diverse as the speech recognition, multimodal anomaly identification during industrial big data, robot manipulation tasks, remote monitoring of patients, and acoustic modeling. To the best of our knowledge, there are two dominant approaches to anomaly identification with time-series using RNNs: Prediction-based approaches and reconstruction-based approaches.

Since the RNNs model is widely applied to time series prediction and reconstruction, we mainly review related works that use RNNs for temporal anomaly detection. RNNs-based Time series prediction has been popularly investigated in recent decades, LSTMs are capable of learning the underlying pattern between past observations and current observations and representing that patterns in the form of learned weights. For instances, the traffic flow prediction with LSTMs is proposed in Shao and Soong and Liu et al. experimental results indicate that the intrinsic feature of LSTMs for capturing long-term dependencies in sequential data such that make it a suitable choice in complex time series prediction. In Sagheer and Kotb, a prediction model is designed by using temporal attention mechanism
on top of stacked LSTMs for multivariate time series prediction and used to predict pollution levels. The excellent prediction performance of LSTMs, makes them an ideal candidate for anomaly detection. In addition, recent solutions of collective anomalies identification have been proposed for mitigating false positive and improving the accuracy, which commonly employ sliding windows to transform a time-series into a set of labels for implementing the predication in a supervised fashion. Subsequently, modeling the prediction errors using multivariate Gaussian Model or other statistical tricks (e.g. mean, medium, and maximum). In Hundman et al.,26 takes LSTMs as prediction model and mitigates false positive by introducing the percentage of decrease of the max prediction error for anomaly detection at each time step. In Bontemps et al.,29 LSTMs are used for detecting collective anomalies in network security domain, in which a prediction model with a single recurrent layer is used and a point anomaly is identified when a prediction error greater that a lower bound.

In Malhotra et al.,36 stacked LSTMs approach (LSTM-AD) and stacked RNNs approach (RNN-AD) using recurrent sigmoid units are employed to identify anomalies by modeling the prediction errors of time series. The model takes only one time step as input variable to predict multiple time steps, maintains LSTM units in a hidden layer are fully connected through recurrent connections, and is trained on normal dataset. Thus each observation in normal sequence should have multiple predictions made at different times in the past, an error value can be computed between the prediction and input. Consequently, error vectors are calculated from the multiple predictions, which are modeled using a multivariate Gaussian distribution to give the likelihood of an anomaly. The same implementation procedure is also applied to detect anomalies in Electrocardiography signals.43 Those results are promising that a valuable reference for anomaly detection of multivariate time series.

Beside that, recent works have been proposed to comparing the reconstruction error induced by deep autoencoders, and demonstrate promising results.44 The reconstruction-based methods assume that anomalies are incompressible and thus cannot be effectively reconstructed from low-dimensional projections, such as LSTM-based Variational Autoencoder (LSTM-VAE),33 Deep Autoencoding Guassian Mixture Model (DAGMM),45 and its variant that the encoder and decoder are designed with LSTM architecture (LSTM-DAGMM), both of them also reported good performance for multivariate anomaly detection.

In this work, we follow the promising success of deep learning-based multivariate anomaly detection and implement the comprehensive comparison of recurrent neural network based autoencoder (AE), stacked Long-short Term Memory (sLSTM), LSTM-based encoder and decoder (LSTM-ED), LSTMs-based Variational Autoencoder (LSTM-VAE), Deep Autoencoder Gaussian Mixture Model (DAGMM), and its variants with LSTM encoder-decoder (LSTM-DAGMM) based on a human-robot collaborative system. The contributions of this paper can be summarized as follows:
- A comprehensive comparison of multimodal anomaly detection using the state-of-the-art deep-learning methods is proposed, which can evaluate the performance for learning the data more effectively under different robot execution phase;
- An intuitive comparison of various modalities combination for anomaly detection based on the state-of-the-art deep-learning methods in a real-world dataset;
- An intuitive comparison of various anomaly bias for anomaly detection based on the state-of-the-art deep-learning methods in a real-world dataset;
- This implementation scheme can be easily used to anomaly monitoring of unmanned production line, machinery, and healthcare.

Learning latent representation of normal multimodal observations

Multimodal time series

With the development of multi-modal sensing fusion technology and the diversity of the environment, except the joint encoders, robots often need to add force/torque sensors and tactile sensors to sense the surrounding environment, so that the robot’s observation is often multimodal. In general, a multidimensional time series with dimension $D$ and length $T$ is represented as follows:

$$Y_{T\times D} = \begin{bmatrix} y_1^{(1)} & y_1^{(2)} & \cdots & y_1^{(D)} \\ y_2^{(1)} & y_2^{(2)} & \cdots & y_2^{(D)} \\ \vdots & \vdots & \ddots & \vdots \\ y_T^{(1)} & y_T^{(2)} & \cdots & y_T^{(D)} \end{bmatrix},$$

where, \( t \) represents a time frame, \( d \) represents a variable, and \( y_d^{(t)} \) denotes the observation of the \( d \)-dimensional variable at the \( t \)-th time step. For ease of writing, define the observation vectors for all dimensions at time \( t \) as \( y^{(t)} \in \mathbb{R}^D \). Hence, the equation (1) is expressed as a matrix with $T\times D$, and the matrix is set as an example of a multidimensional time series, that is,

$$Y_{T\times D} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(T)} \end{bmatrix} = \begin{bmatrix} y_1^{(1)} & \cdots & y_1^{(D)} \\ \vdots & \ddots & \vdots \\ y_T^{(1)} & \cdots & y_T^{(D)} \end{bmatrix}$$

Each row in equation (2) represents the observation at time \( t \), and each column represents the data of a signal. To learn and analyze the multi-dimensional time series based on those following deep learning-based methods.
Autoencoder

The autoencoder is designed to model the multimodal data by integrating multi-layer perception neural networks with three hidden layers, and both the input and output layers have \( D \) units, corresponding to the \( D \) features of the training data, that is, input vectors \( y^{(i)} \in \mathbb{R}^D \) are mapped to desired output vectors \( \hat{y}^{(i)} \in \mathbb{R}^D \) in multi-layer perception neural networks. The number of units in the three hidden layers are chosen experimentally to minimize the average reconstruction error across all training patterns. The anomaly detection is then developed as the reconstruction error of individual observations. Specifically, we assume an anomalous observation \( y^{(i)} \) is detected by the average reconstruction error \( e(y^{(i)}) \) over all \( D \) features by

\[
e(y^{(i)}) = \frac{1}{D} \sum_{d=1}^{D} (y_d^{(i)} - \hat{y}_d^{(i)})^2
\]

To detect anomaly, the trained autoencoder can be used to score each observation with the value of reconstruction error as formulated in equation (3), and then the sort the scores in descending order, the score ranked higher are likely anomalies. Therefore, an anomaly detector based on autoencoder can be developed with a quantitative score based on reconstruction error and determined the threshold with the percentage of sorted scores.

Stacked LSTM

Long Short Term Memory (LSTM) networks can effectively learn underlying temporal and spatial pattern of sequential data with unknown length because of their ability to maintain long term memory. A networks with multiple LSTM would improve the learning ability of temporal dynamics, named stacked LSTM (sLSTM). To address the problem of anomaly detection, the sLSTM is trained on normal observations and used as a predictor over a number of time steps (sliding window). Unlike the autoencoder in Section 3.2, a detector base on sLSTM is integrated with a multivariate Gaussian distribution that used for modeling the prediction error, and then the anomaly can be triggered by assessing the likelihood of testing error.

Specifically, a multimodal time series \( Y_{T \times D} \) is recorded from a robot execution, whose elements \( y^{(i)} \) are multimodal observations with a \( D \)-dimensional vector, and then transfer the time series into supervised fashion using a sliding window \( w \). A prediction model using sLSTM learns to predict the next \( b \) steps with the input variables \( w \times D \), and then modeling the resulting prediction error at each time step by computing the corresponding parameters (mean and covariance) of multivariate Gaussian distribution. In the prediction scheme, with a prediction length of \( b \), the prediction errors of the first \( w + (b - 1) \) observations are ignored, and for time step \( (w + b) < t < (T - w) \) is predicted \( b \) times, the resulting error vector \( e^t \) at time \( t \) can be denoted as \( e^t = [e_1^t, e_2^t, ..., e_b^t] \), the elements represent the errors predicted...
at various sliding window, where \( e_b^{(t)} = [e_{b1}^{(t)}, e_{b2}^{(t)}, ..., e_{bd}^{(t)}] \) is the difference between \( y^{(t)} \) and \( \hat{y}^{(t)} \) of each dimensionality.

To detect anomaly, there are \( b \) prediction error vectors of each observation, and then modeled using multivariate Gaussian distribution \( e^{(t)} \sim \mathcal{N}(\mu, \Sigma) \), thus the likelihood \( L^{(t)} \) of observing an error vector \( \tilde{e}^{(t)} \) can be calculated based on the trained Gaussian model. The threshold \( \rho \) for anomaly detection can be obtained using Kalman filter based dynamic prediction model as describe in literature.\(^46\) Therefore, a testing observation \( \tilde{y}^{(t)} \) is detected as anomaly if \( L^{(t)} < \rho \), otherwise the observation is normal.

**LSTM-ED**

As described in Section 3.2 and 3.3, the autoencoder and LSTM networks can successfully applied to anomaly detection of multimodal observation. However, it remains challenging on those situations that with unpredictable, periodic, anomalies using prediction errors. A LSTMs based encoder-decoder, named LSTM-ED,\(^47\) is proposed for anomaly detection, which attempt to reconstruct the normal dynamics via latent representation learning and uses the reconstruction error to detect anomalies. That is, when given an anomalous instances, it may not be able to reconstruct it well, and hence would lead to higher reconstruction errors compared to the reconstruction errors for the normal instances. According to Section 3.2, a scoring scheme also proposed for the determining anomalies, where the reconstruction error at any time step is used to calculate the likelihood of anomaly at that time. A higher anomaly score indicates a higher likelihood of the point being anomalous.

Specifically, a normal multimodal time series \( Y_{T \times D} \) of length \( T \) is considered, where each observation \( y^{(t)} \in \mathbb{R}^D \) is a \( D \)-dimensional vector at time \( t \). The LSTM-ED network learns a fixed length vector representation \( z^{(t)} \in \mathbb{R}^H \) of the input instances \( y^{(t)} \) and uses this representation to reconstruct the instances \( \hat{y}^{(t)} \) using the current hidden state and the value predicted at the previous time-step, where \( H \) is the number of latent space. The LSTM-ED is trained to minimize the difference between source input \( y^{(t)} \) and reconstruct output \( \hat{y}^{(t)} \) such that the objective

\[
\sum_{Y_{T \times D} \in \mathbb{S}_N} \sum_{t=1}^T \| y^{(t)} - \hat{y}^{(t)} \|^2
\]

where the \( \mathbb{S}_N \) is set of normal training dataset with \( n \) trials .

To detect anomaly, we should observe the anomaly at each time \( t \) by computing the likelihood of reconstruction error \( e^{(t)} \) based on multivariate Gaussian model \( e^{(t)} \sim \mathcal{N}(\mu, \Sigma) \), where the error vector is given by \( e^{(t)} = |y^{(t)} - \hat{y}^{(t)}| \). For all \( n \) instances of normal training, the parameters \( \mu \) and \( \Sigma \) of Gaussian distribution can be estimated using Maximum Likelihood Estimation, and then, for each observation \( y^{(t)} \), the anomaly score is computed by
An anomaly can be determined based on a threshold \( \rho(t) \) of the resulting scores at each time in a supervised setting, that is, if \( s(t) > \rho(t) \), an observation is labeled as anomalous, otherwise normal. Similar to, \(^{48}\) the threshold can be learned by maximizing \( F_\beta = (1 + \beta^2) \times P \times R / (\beta^2 P + R) \) when enough anomalous trials are available, where \( P \) is precision, \( R \) is recall, and \( \beta < 1 \).

**VAE**

A Variational Autoencoder (VAE) is a variant of an AE (described in Section 3.2) rooted in Bayesian inference. \(^{49}\) Similar to the AE formulation, a VAE is able to represent the high-dimensional data \( y \in \mathbb{R}^D \) by introducing a set of latent random variables \( z \) and model the underlying distribution of observations \( p(z) \) such that the implementation can be denoted by \( p(y) = \int p(y|z)p(z)dz \). However, the marginalization \( p(y) \) is computationally intractable since the latent space of \( z \) is unobserved and continuous such that we cannot differentiate the marginal likelihood. Instead, the marginal log-likelihood of an individual observation can be formulated as

\[
\log p(y) = D_{\text{KL}}(q_\phi(z|y)||p_\theta(z)) + \mathcal{L}_{\text{vae}}(\phi, \theta; y)
\]

Where \( D_{\text{KL}} \) is Kullback-Leibler divergence (non-negative) from a prior \( p_\theta(z) \) to the variational approximation \( q_\phi(z|y) \) of \( p(z|x) \), and \( \mathcal{L}_{\text{vae}} \) is the variational lower bound of \( y \) by Jensen’s inequality. The \( \phi \) and \( \theta \) are the parameters of the encoder and decoder, respectively, which optimized by maximizing the lower bound of the log-likelihood \( \mathcal{L}_{\text{vae}} \), written as

\[
\mathcal{L}_{\text{vae}} = - D_{\text{KL}}(q_\phi(z|y)||p_\theta(z)) + \mathbb{E}_{q_\phi(z|y)}[\log p_\theta(x|z)]
\]

Where the first term regularizes the latent variable \( z \) by minimizing the KL divergence and the second term is the reconstruction of \( y \) by maximizing the log-likelihood \( \log p_\theta(y|z) \) with sampling from \( q_\phi(z|x) \). To derive the unknown parameters, we define the posterior distribution as a Gaussian distribution \( q_\phi(z|x) \sim \mathcal{N}(\mu_z, \Sigma_z) \) and the prior as a standard normal distribution \( p_\theta(z) \sim \mathcal{N}(0, 1) \).

Considering the data characteristics during human-robot collaboration, a long short-term memory-based variational autoencoder is present by introducing the temporal dependency of time-series data into a VAE. The feed-forward network in a VAE is replaced with LSTMs, which similar to conventional temporal AEs (Section 3.2) such as aforementioned LSTM-AD (Section 3.3) and LSTM-ED (Section 3.4).

To detect anomaly, we also define an anomaly score \( s(t) \) as the negative log-likelihood of an observation with respect to the reconstructed distribution of the observation through VAE with

\[
s(t) = - \log p(y(t); \mu_\phi(t), \Sigma_\phi(t))
\]
Where, $\mu_{y(t)}$ and $\Sigma_{y(t)}$ are the mean and co-variance of the reconstructed distribution. Therefore, an anomaly detected when $s(t)$ of an observation $y(t)$ is higher than a score threshold $\rho$. A high score indicates an input has not been reconstructed well by the LSTM-VAE model that trained with non-anomalous executions. For the threshold $\rho$ calculation, we refer to literature, the $\rho$ is computed by mapping a multidimensional input $z$ to a scalar using support vector regression $f(\cdot)$ with a radial basis function kernel from a non-anomalous dataset, that is $\rho = f(z)$. To change the detection sensitivity, a constant $c$ can be added to the expected score and the threshold represented as $\rho = f(z) + c$.

**DAGMM**

As described in Section 3.5, the sparse representation of multimodal observation has achieved great success in anomaly detection without human supervision. However, it’s difficult to effectively perform density estimation in the latent space when the dimensionality of input data becomes higher. To address this problem, a Deep Autoencoding Gaussian Mixture Model (DAGMM) for unsupervised anomaly detection is proposed, which combining the dimensionality reduction and density estimation in the latent space for the high-dimensional observations. That is, DAGMM utilizes a deep autoencoder to generate a low-dimensional representation and reconstruction error for each observation, which is further fed into a Gaussian Mixture Model (GMM) for density estimation.

Specifically, a normal multimodal time series $Y_{T \times D}$ of length $T$ is considered, where each observation $y(t) \in \mathbb{R}^D$ is a $D$-dimensional vector at time $t$. The DAGMM network is trained to learn sparse representation $z_r = [z_c, z_r]$ of high-dimensional input and reconstruct from $z_r$, where $z_c$ is the reduced low-dimensional representation learned by the deep autoencoder, $z_r$ includes the features derived from the reconstruction error. Unlike the LSTM-VAE model, the latent space is restricted to be normal distribution $z_r \sim N(0, I)$. Particularly, $z_c$ can be multi-dimensional, considering multiple distance metrics such as absolute Euclidean distance, relative Euclidean distance, cosine similarity, and so on.

To detect anomaly, this model should mainly focus on the novel proposal on estimation network based on GMM with $K$ components that trained on reconstruction errors. For each component that including three kinds of parameters: mixture-component distribution $\Phi_k$, mean $\mu_k$, and covariance $\Sigma_k$, where $k \in \{1, 2, ..., K\}$. Therefore, the likelihood of testing sample can be formulated by

$$L(z) = -\log\left(\sum_{k=1}^{K} \Phi_k \exp\left(-\frac{1}{2}(z - \mu_k)^T \Sigma_k^{-1}(z - \mu_k)\right) \sqrt{2\pi \Sigma_k}\right)$$  \hspace{1cm} (9)

Where $|\cdot|$ denotes the determinant of a matrix. Therefore, to detect anomaly based on the calculated likelihood in equation (9) during the testing phase, where would predict samples of high likelihood as anomalies by a constant threshold. In this
implementation, the threshold is chosen with the top 20% samples of the highest likelihood are marked as anomalies.

Additionally, motivated by the continued success of LSTMs on learning temporal characteristics, a extension of DAGMM using an LSTM-autoencoder as the compression network instead of a neural network autoencoder, named as LSTM-DAGMM, and the scheme of anomaly detection is the same as DAGMM.

Experimental verification and discussion

Experimental setup

To compare the anomaly detection performance of considered methods in a multimodal dataset. We designed a HRC task with six respective movements for picking and placing object into a container ($300\text{mm} \times 250\text{mm} \times 200\text{mm}$) using Baxter robot, named as kitting experiment. Specifically, a human co-worker places a set of six objects marked with Alvar tags (http://wiki.ros.org/ar_track_alvar) on the robot’s reachable region (located in front of the robot) in a one-at-a-time fashion. The detail information of objects is illustrated in Table 1.

The objects may accumulate in a queue in front of the robot once the first object is placed on the table, the robot’s left arm camera identifies the object and the robot’s right arm picks, transports, and places it in a container located to the right of the robot, as shown in Figure 2. After which, the robot appropriately places each of the six objects in different parts of the container.

Multiple sensors should be installed for effectively identifying the unexpected anomalies in such a kitting experiment. Here, the right arm of Baxter robot is equipped with a six degrees of freedom (DoF) Robotiq F/T sensor and two Baxter-standard electric pinching fingers, where each finger is further equipped with a multimodal tactile sensor composed of a $4 \times 7$ taxel matrix that yields absolute pressure values. In addition, Baxter’s left hand camera is placed flexibly in a region that can capture objects in the collection bin with a resolution of $1280 \times 800$ at 1 fps (we are optimizing pose accuracy and lower computational complexity in the system). The use of the left hand camera facilitated calibration and object tracking accuracy.

| Index | Name                     | Material | Weight | Shape (mm$^3$) |
|-------|--------------------------|----------|--------|----------------|
| 1     | Stapler box              | Paper    | 186.9 g| $73 \times 73 \times 113$ |
| 2     | Toolbox (blue)           | Colloid  | 30.8 g | $65 \times 35 \times 110$ |
| 3     | Brush pot                | Metal    | 368.4 g| $70 \times 70 \times 109$ |
| 4     | Bottled drinks (green)   | Plastic  | 1055 g | 1L*            |
| 5     | Bottled drinks (yellow)  | Plastic  | 767.1 g| 1L*            |
| 6     | Ink box                  | Paper    | 119.1 g| $80 \times 80 \times 50$ |

*Irregular shape.

Yang et al. 11
Experimental procedure

The detailed movement implementation is shown in Figure 2. The kitting experiment consists of six individual movements, including (Movement 3): Home → Pre-pick; (Movement 4): Pre-pick → Pick; (Movement 5): Pick → Pre-pick; (Movement 7): Pre-pick → Pre-place; (Movement 8): Pre-place → Place; (Movement 9): Place → Pre-place) that were effectively modeled with DMP and ROS-SMACH (http://www.ros.org/SMACH). One-shot kinesthetic demonstrations were used to train DMP models for each of the Movements of the kitting experiment. Note that a movement’s starting and ending pose can be adapted if necessary thus providing a flexible and generalizable movement encoding procedure. The robot is tasked to pick each one of the objects and place them in the container to its right. The visualization module uses the ALVAR tags to provide a consistent global pose with respect to the base of the robot. The trajectory adaptations will change the lengths...
of movements and increase the variability in the data collection even from nominal executions, resulting in augmenting the difficulty of assessing the sensory data with temporal uncertainty for robust anomaly detection.

**Potential anomalies**

External disturbances may be introduced into such a HRC task for a variety of reasons, as illustrated in Table 2.

| Index | Type of anomalies          | Comment                                                                                                                                 |
|-------|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| 1     | Human Collision (HC)      | A human co-worker to accidentally collide with the robot joints in unexpected ways.                                                       |
| 2     | Tool Collision (TC)       | A human co-worker may also unintentional move packaging objects in ways a robot may not anticipate, and the inevitable error of object pose estimation from vision. Such variations may lead to robot’s end-effector collide with environment. |
| 3     | Object Slip (OS)          | Picked objects may slip from a robot’s gripper.                                                                                         |
| 4     | Wall Collision (WC)       | The robot may also collide with container during transporting and kitting.                                                              |
| 5     | No Object (NO)            | The object missed-grasps because of the cumulative errors from vision and robot trajectory.                                              |
| 6     | False Positive (FP)       | The false-positives from the anomaly detector, which include the system errors, unreachable objects, unfeasible inverse kinematic solutions, unidentifiable objects from the visual system, etc. |

**Multimodal dataset collection and description**

To effectively capture the underlying dynamics for each movement, we tasked five participants as collaborator (one expert user who confidently know this implementation and other four novice users) in our designed kitting experiment. Novice users first learned from the expert to induce anomalies during robot executions, which would aggravate the external uncertainty and increase the modeling difficulties. During data collection, each participant performed one nominal and six anomalous executions by placing the set of six household objects in a one-by-one fashion. Consequently, we ignore the failure executions and totally collected a dataset with 18 nominal executions and 180 anomalous executions, where each anomalous execution has at least one anomaly.

**Dataset partition and labeling.** After collection, we assume a multivariate time series $Y = \{y_1, y_2, ..., y_T\}$ of each robot execution, whose elements can be regard as the input variables of a considered model, where each point $y_t \in \mathbb{R}^D$ in the time series
is an $D$-dimensional vector. An then, we divide the robot execution into several phases (movements) for intuitively comparing the performance of considered models during various situations. The further information of the dataset is described in Table 3. Then, the recorded dataset is partitioned into two categories: training dataset with nominal observations and testing dataset of mixture both abnormal and nominal observations. Then, we use the training dataset to train a considered model and compare the performance in the testing dataset.

**Sensory Preprocessing.** As for our multimodal observation vector $y_t$ originally consists of six force-torque features, six Cartesian velocity features of Baxter’s right end-effector, and 56 tactile features from both the left and right tactile sensors. The multimodal sensory data is recorded online and resampled at $20\text{Hz}$ for guaranteeing temporal synchronization among three kind of sensors and then further preprocessing. In consideration of the redundant features would aggravate computational efficiency and increase false-positive rate (anomaly frequently occur even when robot’s movement is normal), six empirical features extraction are performed on the original observation vector to improve the identification performance.

**Wrench modality.** The raw signals of F/T sensor is denoted by $(f_x, f_y, f_z, t_x, t_y, t_z)$. We wish to extract structural information from the wrench instead of simply using raw values. In this way, we can find signal patterns that may occur across the different DoFs. To this end, we computed the norm of both the force $n_f$ and the torque $n_t$ as features, respectively:

$$n_f = \sqrt{f_x^2 + f_y^2 + f_z^2}, \quad n_t = \sqrt{t_x^2 + t_y^2 + t_z^2}. \quad (10)$$

**Velocity modality.** Similarly, we take the Cartesian linear $(l_x, l_y, l_z)$ and angular $(a_x, a_y, a_z)$ velocities and compute their norm $n_l$ and $n_a$, respectively:

$$n_l = \sqrt{l_x^2 + l_y^2 + l_z^2}, \quad n_a = \sqrt{a_x^2 + a_y^2 + a_z^2}. \quad (11)$$

**Table 3.** The information of each movement during robot kitting experiment with the anomaly bias $"[-1, 1]"$.

| Movement           | Train | Test | Anomalies |
|--------------------|-------|------|-----------|
| Home→pre-pick      | 14,270| 7375 | 240       |
| Pre-pick→pick      | 7694  | 4768 | 920       |
| Pick→pre-pick      | 6554  | 4374 | 1097      |
| Pre-pick→pre-place | 4670  | 5113 | 2777      |
| Pre-place→place    | 653   | 407  | 80        |
| Place→pre-place    | 307   | 194  | 40        |

The nominal observations are labeled with 0, and the anomalous observations are labeled as 1 during training and testing.
**Tactile modality.** Due to the computational cost of processing the tactile sensor’s high dimensionality, we empirically tested a number of features for each tactile sensor, they include: the maximum taxel value, the largest five taxel values, the mean of all taxel values, and the standard deviation for all taxel values. It was the standard deviation, which proved to be the most useful feature for anomaly detection. The standard deviation for each tactile sensor \( s_l \) and \( s_r \) are defined as:

\[
 s_l = \sqrt{\frac{1}{28} \sum_{i=1}^{28} (l_i - \mu_l)}, \quad s_r = \sqrt{\frac{1}{28} \sum_{i=1}^{28} (r_i - \mu_r)}. \tag{12}
\]

where \( \mu_l = \frac{1}{28} \sum_{i=1}^{28} l_i \) and \( \mu_r = \frac{1}{28} \sum_{i=1}^{28} r_i \) are the mean of each tactile sensor respectively. Equations (10)–(12) facilitate the identification of diverging signals during anomalous situations. We empirically concatenate all the valuable features for representing the robot executions both in nominal and anomalous cases.

In our prior knowledge, those extracted features are acting respective roles involving the wrench modality would sense the collision with the environment, the velocity modality would perceive the accidental human collision various in orientation and magnitude as well as the tactile modality would take responsibility to the object slip and no object situation. Therefore, our feature vector \( y_t \) can be randomly formulated with the raw signals and the extracted statistical features from equations (10)–(12). For instance, an evolving extracted features of 10 nominal executions are illustrated in Figure 3.

**Verification and evaluation metrics**

To evaluate the feasibility and efficiency of the considered methods for anomaly detection in the HRC scenarios by assessing the tradeoff between false positive (FPR) and true positive (TPR) rates. We first present all of our detection results as receiver operating characteristic curves (ROC) across six movements in the kitting experiment. Each resulting point in ROC is generated by varying a single parameter \( c \), from our threshold definition. Then, we compute Area Under the Curve (AUC) using the trapezoidal rule for evaluating the whole performance of individual movement.

**Performance comparison**

**Various modalities combinations.** We investigated if multiple modalities improve the detection performance of those deep learning models. Table 4 shows average AUC over modality combinations of various modalities (i.e. force \( f = [f_x, f_y, f_z] \), torque \( t = [t_x, t_y, t_z] \), linear velocity \( v_l = [l_x, l_y, l_z] \), angular velocity \( v_a = [a_x, a_y, a_z] \), tactile \( [s_l, s_r] \) (including the standard deviation of each tactile sensor \( s_l \) and \( s_r \)), etc.), and the six statistical modality \([n_f, n_t, n_l, n_a, s_l, s_r]\) across the various movements. Table 4 shows the average AUC of all the robot movements over combinations of
modalities (i.e. force, torque, velocity, tactile) with considered models. The use of norm magnitudes of all considered modalities $n_f + n_t + n_l + n_a + s_l + s_r$ achieved the best performance for all models. The AUC results indicated that multiple modalities would substantially more effective than uni-modalities in anomaly detection, especially, by adding the norm magnitude. However, it does not indicate that more modalities always enhance detection performance.

Various anomaly bias. In this section, we evaluate the performance with various anomaly bias (Changing the anomaly region that labeled by human during data collection) under the optimal modalities combinations $n_f + n_t + n_l + n_a + s_l + s_r$ that with the best performance of all models. Table 5 shows average AUC under various anomaly bias, the larger the interval, the more anomalous observations, in which the interval $[0, 2]$ have the best performance across all considered models. With this analysis, we can subsequently extract the anomalous samples after $2s$ moment that anomaly detected. That’s, the anomaly data should be persistence and lack of forewarning. Besides that, we can know that the LSTM-DAGMM still outperforms the others across all the various anomaly bias. Additionally, we can confidently extract the anomaly samples during the period of $[0, 2]$ for anomaly diagnosis when anomaly detected with this verification.

![Figure 3](image_url)

**Figure 3.** Illustrates the extracted features of ten nominal executions in the kitting experiment, where different movements are represented in different colors. “Gray” represents the robot is moving to home or pause, we will ignore those situations in this paper; “Red, Green, Blue, Cyan, Magenta, and Yellow” indicate the underlying dynamics in Movement 3, Movement 4, Movement 5, Movement 7, Movement 8, and Movement 9, respectively. We can intuitively assume that almost all the unexpected anomalies can be identified by monitoring the wrench, velocity, and tactile modalities.
Table 4. Anomaly detection performance (the AVERAGE AUC across all the movements) over various modalities combinations with fixed anomaly bias “[-1, 1].”

| Index | Modalities | AE  | sLSTM | LSTM-ED | LSTM-VAE | DAGMM | LSTM-DAGMM | Avg.  |
|-------|------------|-----|-------|---------|----------|-------|-------------|-------|
| 1     | f          | 0.620 | 0.865 | 0.577   | 0.853    | 0.898 | 0.989       | 0.800 |
| 2     | vl         | 0.711 | 0.882 | 0.738   | 0.782    | 0.873 | 0.993       | 0.830 |
| 3     | f + t      | 0.718 | 0.889 | 0.635   | 0.831    | 0.736 | 0.995       | 0.800 |
| 4     | vl + va    | 0.760 | 0.898 | 0.763   | 0.746    | 0.610 | 0.993       | 0.795 |
| 5     | nf + nl    | 0.765 | 0.909 | 0.772   | 0.835    | 0.497 | 0.998       | 0.796 |
| 6     | nt + na    | 0.844 | 0.904 | 0.820   | 0.825    | 0.899 | 0.878       | 0.862 |
| 7     | sl + sr    | 0.596 | 0.755 | 0.525   | 0.921    | 0.806 | 0.989       | 0.765 |
| 8     | f + sl + sr| 0.670 | 0.898 | 0.624   | 0.884    | 0.816 | 0.999       | 0.815 |
| 9     | t + sl + sr| 0.727 | 0.930 | 0.666   | 0.847    | 0.784 | 0.992       | 0.824 |
| 10    | vl + sl + sr| 0.737 | 0.946 | 0.706   | 0.793    | 0.777 | 0.995       | 0.826 |
| 11    | f + t + vl + va| 0.823 | 0.974 | 0.802   | 0.740    | 0.736 | 0.986       | 0.843 |
| 12    | nf + nl + sl + sr| 0.782 | 0.934 | 0.738   | 0.857    | 0.978 | 0.999       | 0.881 |
| 13    | t + vl + va + sl + sr| 0.829 | 0.974 | 0.815   | 0.752    | 0.769 | 0.859       | 0.833 |
| 14    | nf + nt + nl + na + sl + sr| 0.850 | 0.970 | 0.895   | 0.831    | 0.829 | 0.996       | 0.895 |
| 15    | f + t + vl + va + nf + nt + nl + na + sl + sr| 0.856 | 0.967 | 0.849   | 0.741    | 0.769 | 0.991       | 0.862 |
|       | The average AUC across modalities | 0.752 | 0.913 | 0.728   | 0.816    | 0.785 | 0.977       |       |

Where f, t, v, s_l, s_r represent force of F/T sensor, torque of F/T sensor, velocity of robot end-effector, the standard deviation of tactile signals. Those underlined values represent the best performance of models across various modalities. The bold values represent the best performance.
Table 5. Anomaly detection performance (the AVERAGE AUC across all the movements) over various anomaly bias with considered models, where “$[-1, 1]$” denotes the observations located in the anomalous range before/after 1 s of the labeled moment by human during data collection, and “$[0, 0]$”.

| Index | Anomaly Bias | AE   | sLSTM | LSTM-ED | LSTM-VAE | DAGMM | LSTM-DAGMM | Avg  |
|-------|--------------|------|-------|---------|----------|-------|-------------|------|
| 1     | $[-1, 0]$    | 0.773| 0.909 | 0.688   | 0.782    | 0.646 | 0.963       | 0.794|
| 2     | $[0, 1]$     | 0.848| 0.991 | 0.822   | 0.866    | 0.732 | 0.989       | 0.875|
| 3     | $[-1, 1]$    | 0.850| 0.970 | 0.895   | 0.831    | 0.829 | 0.996       | 0.895|
| 4     | $[-2, 0]$    | 0.695| 0.895 | 0.686   | 0.835    | 0.848 | 0.985       | 0.824|
| 5     | $[0, 2]$     | 0.803| 0.980 | 0.780   | 0.917    | 0.967 | 0.989       | 0.906|
| 6     | $[-2, 1]$    | 0.788| 0.952 | 0.771   | 0.845    | 0.957 | 0.994       | 0.884|
| 7     | $[-2, 2]$    | 0.795| 0.952 | 0.773   | 0.859    | 0.701 | 0.994       | 0.846|
| 8     | $[-3, 3]$    | 0.833| 0.866 | 0.840   | 0.873    | 0.899 | 0.930       | 0.874|
| The average AUC across bias | 0.798 | 0.939 | 0.782 | 0.851 | 0.822 | **0.980** |

Those underlined values represent the best performance of models across various anomaly bias.

*Without any anomalous observation marked as anomaly.

The bold values represent the best performance.
We finally present all of our anomaly detection results with optimal modalities combination with \( nf + nt + nl + na + sl + sr \) and the anomaly bias with \([0, 2]\) for six movements, respectively, using receiver operating characteristic curves (ROC) across six movements in the kitting experiment, as shown in Figure 4.

### Conclusion

In this paper, a comprehensive comparison of six representative unsupervised methods for multimodal anomaly detection, including recurrent neural network based autoencoder (AE), stacked Long-short Term Memory (sLSTM), LSTM-based encoder and decoder (LSTM-ED), LSTMs-based Variational Autoencoder (LSTM-VAE), Deep Autoencoder Gaussian Mixture Model (DAGMM), and its variants with LSTM encoder-decoder (LSTM-DAGMM). Experimental verification is performed in a multimodal dataset of self-designed human-robot collaborative task and results indicate that

- Intuitively, the norm magnitude of modality would model the underlying dynamics and depress the noise, which effectively improve the anomaly
detection performance; RNN-based deep learning architecture should improve the modeling capability of time-dependent observations.

- Multiple modalities would substantially more effective than uni-modalities in anomaly detection. However, it does not indicate that more modalities always enhance detection performance;
- LSTM-DAGMM outperforms all the other models for multimodal anomaly detection, which can be easily used to anomaly monitoring of healthcare, autonomous production line, etc;
- For the implementation procedure, reconstruction error-based methods outperform the prediction error-based methods during multimodal anomaly detection.

We believe that this comprehensive comparison will offer a detailed knowledge about the state-of-the-art of recent research in the field of deep learning-based multimodal anomaly detection, and help researchers to pursue research to address open issues and existing challenges in this direction.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work is supported by Guangdong Province Key Areas R&D Program (Grant No. 2019B090919002, 2020B090925001), Guangdong Provincial Key Laboratory of Electronic Information Products Reliability Technology (2017B030314151), Guangzhou Basic and Applied Basic Research Project (Grant No. 202002030237), GDAS’ Project of Thousand doctors(post-doctors) Introduction (2020GDASYL-20200103128), Foshan Key Technology Research Project (Grant No. 1920001001148), Foshan Innovation and Entrepreneurship Team Project (Grant No. 2018IT100173), Guangzhou Science Research Plan Major Project (Grant No. 201804020095), Guangzhou Science and Technology Plan Project (Grant No. 201803010106), Guangdong Province International Cooperation Project of Science and Technology (Grant No. 2019A050510040), National Science Foundation of China (Grant No. 61950410758).

**ORCID iD**

Hongmin Wu https://orcid.org/0000-0001-8517-5296

**References**

1. Berg J and Lu S. Review of interfaces for industrial human-robot interaction. *Curr Rob Rep* 2020; 1(2): 27–34.
2. Matheson E, Minto R, Zampieri EG, et al. Human–robot collaboration in manufacturing applications: a review. *Robotics* 2019; 8(4): 100.
3. Ajoudani A, Zanchettin AM, Ivaldi S, et al. Progress and prospects of the human–robot collaboration. *Auton Rob* 2018; 42(5): 957–975.

4. Maderna R, Poggiali M, Zanchettin AM, et al. An online scheduling algorithm for human-robot collaborative kitting. In: 2020 *IEEE international conference on robotics and automation (ICRA)*, Paris, France, 31 May–31 August 2020, pp.11430–11435. New York: IEEE.

5. Wu H, Guan Y and Rojas J. A latent state-based multimodal execution monitor with anomaly detection and classification for robot introspection. *Appl Sci* 2019; 9(6): 1072.

6. Nakamura A, Nagata K, Harada K, et al. Error recovery using task stratification and error classification for manipulation robots in various fields. In: 2013 *IEEE/RSJ international conference on intelligent robots and systems*, Tokyo, Japan, 3–7 November 2013, pp.3535–3542. New York: IEEE.

7. Srikanth T, Branch P, Jin J, et al. A comprehensive survey of anomaly detection techniques for high dimensional big data. *J Big Data* 2020; 7(1): 1–30.

8. Park D, Erickson Z, Bhattacharjee T, et al. Multimodal execution monitoring for anomaly detection during robot manipulation. In: 2016 *IEEE international conference on robotics and automation (ICRA)*, Stockholm, Sweden, 16–21 May 2016, pp.407–414. New York: IEEE.

9. Ijspeert AJ, Nakanishi J, Hoffmann H, et al. Dynamical movement primitives: learning attractor models for motor behaviors. *Neural Comput* 2013; 25(2): 328–373.

10. Calinon S. Learning from demonstration (programming by demonstration). In: Ang, M.H., Khatib, O. and Siciliano, B (eds.) *Encyclopedia of robotics* 2018. Berlin: Springer. 2018: 1–8.

11. Huang Y, Rozo L, Silvério J, et al. Kernelized movement primitives. *Int J Rob Res* 2019; 38(7): 833–852.

12. Aryal S, Santosh K and Dazeley R. usfAD: a robust anomaly detector based on unsupervised stochastic forest. *Int J Mach Learn Cybern* 2021; 12: 1137–1150.

13. Khalastchi E, Kalech M, Kaminka GA, et al. Online data-driven anomaly detection in autonomous robots. *Knowl Inf Syst* 2015; 43(3): 657–688.

14. Stojanovic L, Dinic M, Stojanovic N, et al. Big-data-driven anomaly detection in industry (4.0): an approach and a case study. In: 2016 *IEEE international conference on big data (big data)*, Washington, DC, 5–8 December 2016, pp.1647–1652. New York: IEEE.

15. Sun M and Zhang J. Data-driven anomaly detection in modern power systems. In: *Security of cyber-physical systems: Vulnerability and Impact*, ch. 7. Cham: Springer International Publishing, 2019, pp.131–143.

16. Alfeo AL, Cimino MG, Manco G, et al. Using an autoencoder in the design of an anomaly detector for smart manufacturing. *Pattern Recognit Lett* 2020; 136: 272–278.

17. Thudumu S, Branch P, Jin J, et al. A comprehensive survey of anomaly detection techniques for high dimensional big data. *J Big Data* 2020; 7(1): 1–30.

18. Habeeb RAA, Nasaruddin F, Gani A, et al. Real-time big data processing for anomaly detection: a survey. *Int J Inf Manage* 2019; 45: 289–307.

19. DiLello HBE, De Laet T. Bruyninckx H. HDP-HMM for abnormality detection in robotic assembly. In: *Proceedings of NIPS workshop on Bayesian nonparametric models for reliable planning and decision making under uncertainty*, Nevada, USA, 3–6 December 2012, pp.131–157.

20. Di Lello E, Klotzbücher M, De Laet T, et al. Bayesian time-series models for continuous fault detection and recognition in industrial robotic tasks. In: 2013 *IEEE/RSJ
international conference on intelligent robots and systems, Tokyo, Japan, 3–7 November 2013, pp.5827–5833. New York: IEEE.

21. Ahmad S, Lavin A, Purdy S, et al. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* 2017; 262: 134–147.

22. Wang X, Lin J, Patel N, et al. Exact variable-length anomaly detection algorithm for univariate and multivariate time series. *Data Min Knowl Discovery* 2018; 32(6): 1806–1844.

23. Pang G, Shen C, Cao L, et al. Deep learning for anomaly detection: a review. *ACM Comput Surv* 2021; 54(2): 1–38.

24. Karadayı Y, Aydin MN and Öğrenci AS. Hybrid deep learning framework for unsupervised anomaly detection in multivariate spatio-temporal data. *Appl Sci* 2020; 10(15): 5191.

25. Hawkins S, He H, Williams G, et al. Outlier detection using replicator neural networks. In: *2002 International conference on data warehousing and knowledge discovery*, Berlin, Germany, 4–6 September 2002, pp.170–180. New York: Springer.

26. Hundman K, Constantinou V, Laporte C, et al. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, London, United Kingdom, 19–23 August, 2018, pp.387–395. New York: ACM.

27. Graves A, Mohamed AR and Hinton G. Speech recognition with deep recurrent neural networks. In: *2013 IEEE international conference on acoustics, speech and signal processing*, Vancouver, BC, 26–31 May 2013, pp. 6645–6649. New York: IEEE.

28. Taylor A, Leblanc S and Japkowicz N. Anomaly detection in automobile control network data with long short-term memory networks. In: *2016 IEEE international conference on data science and advanced analytics (DSAA)*, Montreal, QC, 17–19 October 2016, pp.130–139. New York: IEEE.

29. Bontemps L, McDermott J, Le-Khac NA, et al. Collective anomaly detection based on long short-term memory recurrent neural networks. In: *2017 International conference on future data and security engineering*, Quy Nhon, Vietnam, 25–27 November, 2017, pp.141–152. New York: Springer.

30. Ghosh S, Pal A, Jaiswal S, et al. Segfast-v2: Semantic image segmentation with less parameters in deep learning for autonomous driving. *Int J Mach Learn Cybern* 2019; 10(11): 3145–3154.

31. Hochreiter S and Schmidhuber J. Long short-term memory. *Neural Comput* 1997; 9(8): 1735–1780.

32. Zhou X, Hu Y, Liang W, et al. Variational lstm enhanced anomaly detection for industrial big data. *IEEE Trans Ind Inf* 2020; 17(5): 3469–3477.

33. Park D, Hoshi Y and Kemp CC. A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. *IEEE Rob Autom Lett* 2018; 3(3): 1544–1551.

34. Saini R, Kumar P, Kaur B, et al. Kinect sensor-based interaction monitoring system using the blstm neural network in healthcare. *Int J Mach Learn Cybern* 2019; 10(9): 2529–2540.

35. Sak H, Senior A and Beaufays F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In: *2014 15th annual conference of the international speech communication association*. Singapore, 14–18 September, 2014, pp.338-342. New York: Springer.
36. Malhotra P, Vig L, Shroff G, et al. Long short term memory networks for anomaly detection in time series. In: 2015 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges, Belgium, 22–24 April, 2015, Proceedings. pp. 89–94. Presses universitaires de Louvain.
37. Nguyen VQ, Van Ma L and Kim J. Lstm-based anomaly detection on big data for smart factory monitoring. J Digital Contents Soc 2018; 19(4): 789–799.
38. Li D, Chen D, Jin B, et al. Mad-gan: Multivariate anomaly detection for time series data with generative adversarial networks. In: 2019 International conference on artificial neural networks, Munich, Germany, 17–19 September 2019, pp.703–716. New York: Springer.
39. Husein AM, Arsyal M, Sinaga S, et al. Generative adversarial networks time series models to forecast medicine daily sales in hospital. SinkrOn 2019; 3(2): 112–118.
40. Shao H and Soong BH. Traffic flow prediction with long short-term memory networks (lstms). In: 2016 IEEE region 10 conference (TENCON), Singapore, 22–25 November 2016, pp.2986–2989. New York: IEEE.
41. Liu L, Zhen J, Li G, et al. Dynamic spatial-temporal representation learning for traffic flow prediction. IEEE Trans Intell Trans Syst 2020; 13(6): 1–15.
42. Sagheer A and Kotb M. Unsupervised pre-training of a deep lstm-based stacked autoencoder for multivariate time series forecasting problems. Sci Rep 2019; 9(1): 1–16.
43. Chauhan S and Vig L. Anomaly detection in ecg time signals via deep long short-term memory networks. In: 2015 IEEE international conference on data science and advanced analytics (DSAA), Paris, France, 19–21 October 2015, pp.1–7. New York: IEEE.
44. Shizhen C, and Bo D and Liangpei Z. A Sparse Autoencoder Based Hyperspectral Anomaly Detection Algorithm Using Residual of Reconstruction Error, IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019, pp. 5488–5491, doi: 10.1109/IGARSS.2019.8898697.
45. Zong B, Song Q, Min MR, et al. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In: 2019 International conference on learning representations, Vancouver, Canada, 30 April–3 May 2018, pp.703–716. New York: Springer.
46. Hayton P, Utete S, King D, et al. Static and dynamic novelty detection methods for jet engine health monitoring. Philos Trans R Soc A 2007; 365(1851): 493–514.
47. Saurav S, Malhotra P, TV V, et al. Online anomaly detection with concept drift adaptation using recurrent neural networks. In: CoDS-COMAD ’18: Proceedings of the acm india joint international conference on data science and management of data, January 2018, pp.78–87. ACM.
48. Malhotra P, Ramakrishnan A, Anand G, Vig L, Agarwal P, Shroff G. LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection. CoRR abs/1607.00148 2016.
49. Kingma DP and Welling M. Stochastic gradient vb and the variational auto-encoder. In: 2014 international conference on learning representations, Banff, Canada, 14–16 April 2014, pp.13–19. New York: Springer.

Author biographies

Lin Yang, senior engineer, product certification factory auditors, IECEE peer assessor, graduated from HUST with a bachelor’s degree in engineering in 1988 and has been working at the China Electronic Product Reliability and Environmental Testing Institute at the same year. Mainly engaged in basic theory research of product quality reliability and test testing.
technology, and total solution for quality. From 2000 to 2019 as the director of the quality testing center, currently as the director of equipment and system research unit.

Wu Yan received the M.E. degree in mechanical engineering, Guangdong University of Technology, Guangzhou, China, in 2016. Since then, he has been a Research Assistant at the Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangzhou. His current research interests include object recognition, anomaly detection, machine learning, and Robotics.

Hongmin Wu received the Ph.D. degree in mechanical engineering, Guangdong University of Technology, Guangzhou, China, in 2019. He is currently a Postdoctoral Fellow at the Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangzhou. His research interests lie in the domain of multimodal perception, robot skill learning, learning from demonstration, robot introspection as well as variational inference.