



ARTICLE

Defect Detection Method for Hoist Wire Rope Based on Flow Model Feature Embedding

Tian Ma^{1,2}, Tong An^{1,*}, Jiahao Wang¹ and Ruping Zhang¹

¹College of Artificial Intelligence & Computer Science, Xi'an University of Science and Technology, Xi'an, China

²School of Communication and Information Engineering, Xi'an University of Science and Technology, Xi'an, China

*Corresponding Author: Tong An. Email: 24208208121@stu.xust.edu.cn

Received: 09 June 2026; Accepted: 27 July 2026; Published: 22 October 2026

ABSTRACT: Mine hoist wire rope video inspection is hindered by low illumination, dust interference, motion blur, and vibration, which obscure small defects and destabilize frame-level responses. Given limited defect annotations, this study develops an unsupervised anomaly detection framework based on normalizing flows and trained only with defect-free videos. The framework first applies a Mine Environment Perception Enhancement Module (MEPEM) to mitigate low light degradation and motion blur before spatial anomaly modeling. It then uses a multi-scale detection network based on normalizing flows to embed enhanced features at different resolutions and compute anomaly scores for broken wires, wear, and corrosion. A Transformer temporal consistency module models anomaly score sequences over consecutive frames to suppress transient false alarms caused by illumination fluctuations and dust disturbance. Experiments on a mine hoist wire rope video dataset show that the method achieves a frame-level area under the receiver operating characteristic curve (AUC-ROC) of 93.4% with 60.1 M parameters and an inference speed of 25.1 frames per second (FPS). Compared with the evaluated models based on normalizing flows, it improves AUC-ROC by 0.5–4.5 percentage points and reduces the false alarm rate to 8.4%. These results indicate that the proposed framework is feasible for wire rope video detection under degraded underground imaging conditions.

KEYWORDS: Wire rope defect detection; normalizing flows; low light video enhancement; temporal consistency; transformer

1 Introduction

Machine vision and deep learning methods are widely used for industrial defect detection, including intelligent manufacturing [1], printed circuit boards [2], steel manufacturing [3], deep convolutional inspection models [4], and smart surface inspection [5]. Recent studies further emphasize structure preservation under complex industrial textures [6] and lightweight feature downsampling for ultrasound welding defect inspection [7]. In mine hoists, wire ropes bear load under sustained stress and wear, making them vulnerable to broken wires, corrosion, and surface wear [8]. Real time inspection supports mine safety and intelligent equipment monitoring.

Unlike conventional static industrial inspection, mine hoist wire rope monitoring is usually performed on continuous video. Video frames record surface texture and temporal evolution, but operating mine environments introduce low illumination, dust, flicker, rapid motion, blur, occlusion, and background interference [9]. These factors weaken subtle defect features and disturb the distribution of normal samples. Because wire rope defects are small, irregular, and multi-scale, methods based on individual frames or single scales often produce unstable responses and false alarms.

Earlier nondestructive wire rope inspection studies include optical projection with machine vision for rope arrangement inspection [10]. Such methods are useful in some settings, but underground deployment is still constrained by limited installation space, equipment cost, and sensitivity to harsh operating conditions. Recent vision studies have examined mine hoist wire rope appearance recognition, traction rope defect detection using You Only Look Once (YOLO) [11], and machine vision surface inspection [12]. However, supervised methods require large annotated defect datasets. In actual mine hoist operation, severe defects are usually handled quickly, making sufficient real anomalous samples difficult to collect.

By relying primarily on defect-free samples, unsupervised anomaly detection can reduce dependence on defect annotations in industrial inspection [13]. In this study, “unsupervised” denotes optimization without using defect samples for supervision: defective samples, defect category labels, bounding boxes, and pixel-level masks are not used for training, and the normal feature distribution is learned from defect-free videos. Reconstruction methods use autoencoder feature differences [14], reconstruction embeddings [15], or realistic synthetic anomalies [16] to distinguish abnormal regions. Feature distillation methods include low latency anomaly localization [17]. Probabilistic feature mapping methods model normal feature distributions through pretrained deep features [18], patch embedding with density estimation [19], or anomaly segmentation based on pretrained networks [20]. In degraded mine videos, however, blurred defect boundaries can still cause distribution shifts, inaccurate localization, and increased computational cost.

These limitations are particularly relevant to wire rope videos. Reconstruction methods may smooth subtle broken wires or reconstruct anomalous textures as normal, weakening localization boundaries. Distillation methods depend on the teacher representation and can be sensitive to appearance shifts caused by illumination or blur. Density estimation methods provide a clear probabilistic interpretation, but their feature transformations may become costly when surface details at high resolution and multiple scales must be preserved.

Anomaly detection models based on normalizing flows estimate feature likelihoods through invertible transformations. DifferNet introduced defect detection with normalizing flows for industrial anomaly localization [21]. CS-Flow models feature distributions across scales [22], whereas CFlow-AD uses conditional normalizing flows for localization [23]. FastFlow targets efficient anomaly localization based on normalizing flows [24]. PatchCore uses a representative memory bank of normal patch features for industrial anomaly detection and localization [25]. Other extensions introduce invertible blocks with attention [26], alternating base distribution updating [27], and pyramid normalizing flows for localization at high resolution [28]. However, most models in this group are designed for static images, make limited use of interframe dependencies, and become costly when spatial details at high resolution are modeled in continuous mine videos. Industrial video anomaly detection has recently gained attention, as reflected by the Industrial Process Anomaly Detection (IPAD) dataset [29]. Recent surveys discuss anomaly modeling in industrial inspection contexts [30]. Transformer video anomaly studies highlight temporal context [31], and recent reviews discuss video anomaly detection based on deep learning [32]. These approaches mainly target scene anomalies, with limited focus on subtle wire rope surface defects.

Therefore, wire rope video detection requires spatial anomaly modeling to locate small surface defects and temporal stability to separate persistent defects from short lived disturbances, motivating feature embedding with normalizing flows and sequence level consistency.

Mine hoist wire rope video detection faces three coupled challenges: degraded inputs caused by low light and motion blur, subtle multi-scale defect morphology, and transient video noise that can produce false alarms. To address these issues, this study proposes a video anomaly detection method trained only with normal samples and based on feature embedding with normalizing flows. The improved multi-scale defect detection network based on normalizing flows (ADFN) combines multi-scale feature extraction,

flow mapping at each scale, and cross scale anomaly score fusion to model elongated rope textures and fine defect scales. A causal Transformer at the score level applies bounded residual refinement to ADFN frame scores, reducing transient peaks without changing the spatial detector, while the Mine Environment Perception Enhancement Module (MEPEM) provides degradation compensation tailored to the task before spatial modeling. This design aims to improve defect representation and detection stability while maintaining inference speed suitable for online monitoring.

2 Method Design

2.1 Overall Framework Design

Fig. 1 illustrates the proposed unsupervised anomaly detection framework for wire rope video defect detection. During training, model optimization uses defect-free videos and excludes defective videos. The framework consists of a Mine Environment Perception Enhancement Module (MEPEM) for degradation compensation and a defect detection network based on multi-scale normalizing flows (ADFN) for spatial anomaly modeling.

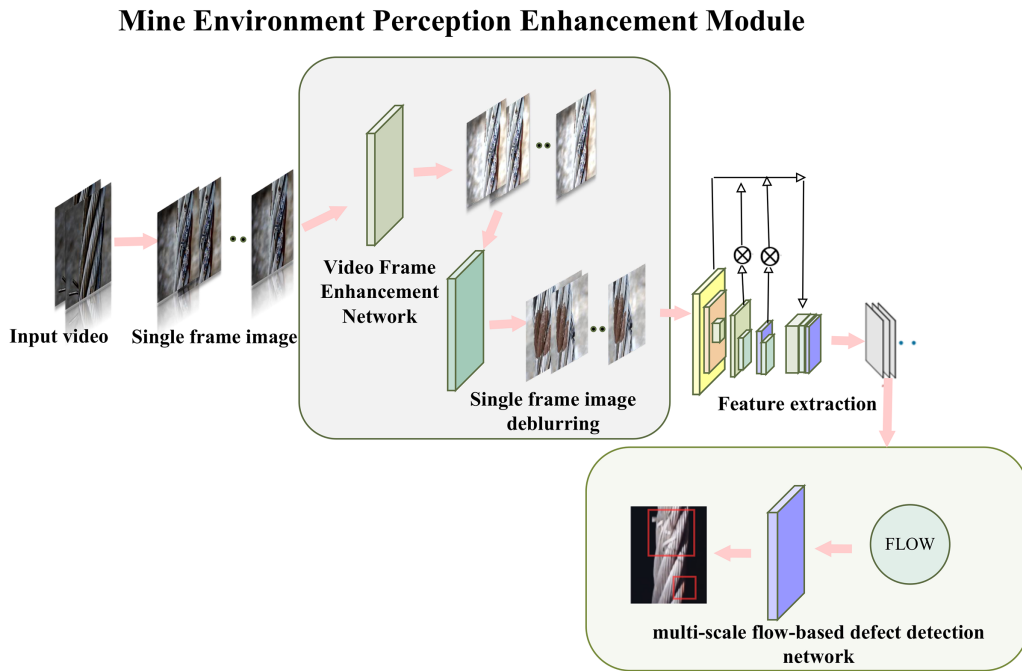


Figure 1: Framework of the video defect detection network based on the multi-scale flow model.

MEPEM first maps degraded frames into a clearer feature space by combining the Video Frame Enhancement Network (VFE) for low light enhancement and the Single Frame Image Deblurring network (SFID) for motion deblurring. ADFN then embeds enhanced multi-scale features into a latent probability distribution with normalizing flows, and the resulting negative log likelihood (NLL) provides frame-level anomaly scores. The Transformer module refines consecutive frame scores to suppress transient fluctuations caused by dust, illumination variation, and vibration, completing the pipeline for degradation compensation, spatial evidence extraction, and temporal stabilization.

2.2 Mine Environment Perception Enhancement Module (MEPEM)

Mine hoist wire rope videos are commonly degraded by low illumination, dust, and rapid motion. A degraded frame I is modeled as a clear image J affected by illumination degradation τ , motion blur kernel K , and environmental noise n , as shown in Eq. (1). MEPEM estimates the inverse compensation process and recovers frame sequences that better preserve wire rope surface structures.

$$I = (J * K) \cdot \tau + n \quad (1)$$

2.2.1 Video Frame Enhancement Network (VFE)

Video Frame Enhancement (VFE) adjusts brightness and contrast in low light frames (Fig. 2) and is related to enhancement based on unsupervised Retinex and exposure fusion [33]. Instead of global histogram equalization, VFE uses a Curve Estimation Network (CEN) to predict nonlinear enhancement parameters for each image. A single enhancement mapping is defined in Eq. (2).

$$LE(I(x); \alpha) = I(x) + \alpha I(x)(1 - I(x)) \quad (2)$$

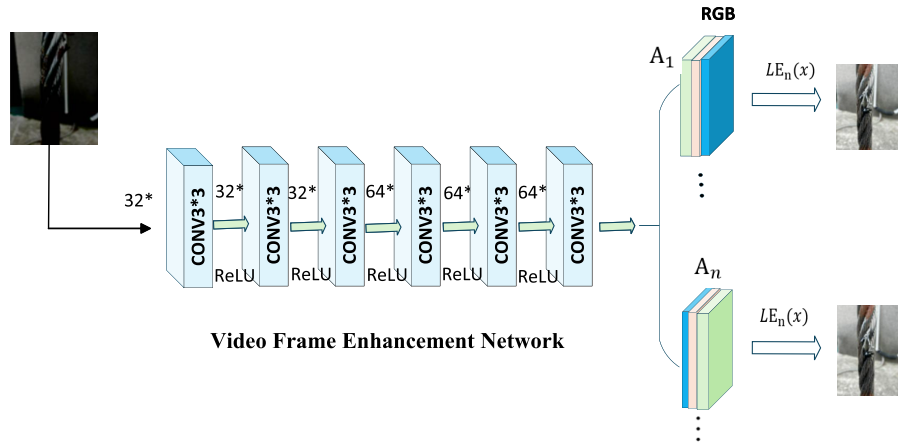


Figure 2: Video frame enhancement network with symmetric feature fusion.

Here, LE denotes the enhancement curve, x is the pixel location, $I(x)$ is the normalized intensity, and $\alpha \in [-1, 1]$ controls exposure. The mapping is applied per pixel and independently to each red, green, and blue (RGB) channel, preserving color while limiting saturation. To cover a wider low light dynamic range, VFE applies the recursive enhancement in Eq. (3).

$$LE_n(x) = LE_{n-1}(x) + \alpha_n LE_{n-1}(x)(1 - LE_{n-1}(x)) \quad (3)$$

The iteration number n controls the curve shape; $n = 1$ reduces to Eq. (2), and $n = 8$ is used in this study. To avoid insufficient or excessive local enhancement from a single global α , a parameter map $A_n(x)$ is adopted for each pixel.

$$LE_n(x) = LE_{n-1}(x) + A_n(x) LE_{n-1}(x)(1 - LE_{n-1}(x)) \quad (4)$$

The predicted A_n limits noise amplification in dark regions and detail loss in overexposed regions, providing more stable inputs for modeling with normalizing flows.

The CEN is a lightweight network with seven convolutional layers and symmetric feature fusion (Fig. 2). It extracts 32-channel features, concatenates paired intermediate features to form the 64-channel fusion stage maps, and predicts a 24-channel parameter map for eight RGB enhancement curve groups. All convolutional layers use 3×3 kernels with stride 1 and padding 1; rectified linear unit (ReLU) is used in the hidden layers and Tanh is applied to the output. Without reference images, VFE is optimized by the composite loss in Eq. (5), which combines spatial consistency, exposure control, color constancy, and total variation regularization on the enhancement curve parameter map.

$$L_{VFE} = \lambda_{spa} L_{spa} + \lambda_{exp} L_{exp} + \lambda_{col} L_{col} + \lambda_{tv} L_{tv}, \quad (5)$$

$$\lambda_{spa} = 1, \quad \lambda_{exp} = 10, \quad \lambda_{col} = 5, \quad \lambda_{tv} = 200.$$

Here, L_{tv} is a total variation regularization term applied to the predicted enhancement curve parameter map A . It penalizes abrupt spatial variations in adjacent curve parameters:

$$L_{tv} = \frac{1}{N_h} \sum \|A_{i+1,j} - A_{i,j}\|_2^2 + \frac{1}{N_w} \sum \|A_{i,j+1} - A_{i,j}\|_2^2, \quad (6)$$

where N_h and N_w denote the numbers of valid adjacent parameter pairs in the vertical and horizontal directions, respectively. Each loss component is averaged over its corresponding spatial or channel units before weighting, and the same coefficients are used in all VFE training experiments.

The spatial consistency loss L_{spa} preserves relative differences between neighboring input and enhanced regions:

$$L_{spa} = \frac{1}{K} \sum_{i=1}^K \sum_{j \in \Omega(i)} (|Y_i - Y_j| - |I_i - I_j|)^2 \quad (7)$$

Here, K is the number of local regions, $\Omega(i)$ is the four-neighborhood of region i , and Y and I are the enhanced and input images. Minimizing this loss preserves wire rope edges while adjusting brightness.

The exposure control loss constrains local brightness around a reference exposure level:

$$L_{exp} = \frac{1}{M} \sum_{k=1}^M |Y_k - E| \quad (8)$$

Here, M is the number of nonoverlapping 16×16 regions, Y_k is the average intensity of region k , and E is set to 0.5 within $[0.4, 0.7]$.

Based on the gray world assumption, color constancy loss L_{col} reduces channel bias in the enhanced output:

$$L_{col} = \sum_{\forall (p,q) \in \varepsilon} (J^p - J^q)^2, \quad \varepsilon = \{(R, G), (R, B), (G, B)\} \quad (9)$$

Here, J^p is the average intensity of channel p , and (p, q) denotes a channel pair. These losses jointly guide brightness, exposure, smoothness, and color compensation in VFE.

2.2.2 Mine Motion Deblurring and Denoising Network

The Single Frame Image Deblurring (SFID) network reduces motion blur using a conditional generative adversarial framework (Fig. 3). The generator G adopts an encoder–decoder structure with deep residual

connections and nine bottleneck residual blocks to preserve high frequency defect details. The patch-based generative adversarial network (PatchGAN) discriminator D evaluates local patch realism. SFID combines content and adversarial losses with $\lambda = 100$:

$$L = L_{advloss} + \lambda L_{content} \quad (10)$$

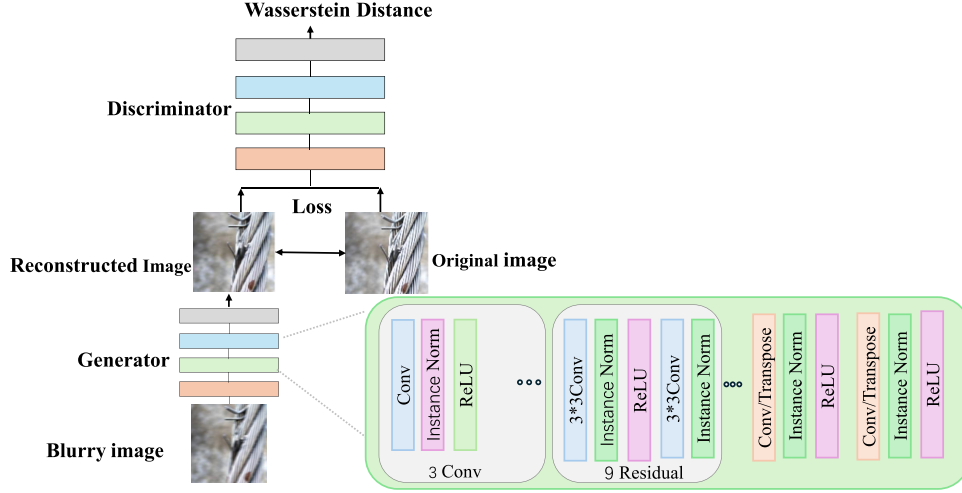


Figure 3: SFID network architecture.

Here, λ balances the two loss terms. A stabilized adversarial training scheme is used for convergence, and the adversarial loss is:

$$L_{advloss} = \sum_{n=1}^N [-D_{\theta_D}(G_{\theta_G}(I^B))] \quad (11)$$

Here, I^B is the blurred image, D_{θ_D} is the discriminator, and G_{θ_G} is the generator. The content loss is computed using mean squared error (MSE) over 19-layer Visual Geometry Group (VGG19) features:

$$L_{content} = \text{MSE}(VGG_{i,j}(I^S) - VGG_{i,j}(G_{\theta_G}(I^B))) \quad (12)$$

Here, $VGG_{i,j}$ is the feature map from the j -th layer of the i -th block in VGG19. These losses guide SFID to reduce motion blur while preserving high frequency rope details.

2.3 Improved Multi-Scale Defect Detection Network Based on Normalizing Flows (ADFN)

After MEPEM, the defect detection network based on normalizing flows (ADFN) performs multi-scale spatial anomaly modeling. It learns the defect-free feature distribution at multiple resolutions for broken wire, wear, and corrosion detection, and its local defect scores are aggregated into frame-level scores for subsequent Transformer temporal modeling.

As shown in Fig. 4, ADFN contains a Multi-Scale Feature Extraction (MSFE) module, a Multi-Scale Flow Module (MSFM), and a Multi-Scale Fusion (MSF) output module. Because wire rope defects range from subtle scratches to larger damaged areas, features are sampled from the end of each stage instead of enforcing a single shared receptive field setting. The sampled maps are $\{z_i \in \mathbb{R}^{D_i \times H_i \times W_i}\}_{i=1}^L$, where $L = 3$

retains fine details with manageable computation. Average pooling with kernel size 3 and stride 2 produces $Y_i = \text{AvgPooling}(z_i)$, $i = 1, 2, 3$, where $Y_i \in \mathbb{R}^{D_i \times \frac{H_i}{2} \times \frac{W_i}{2}}$.

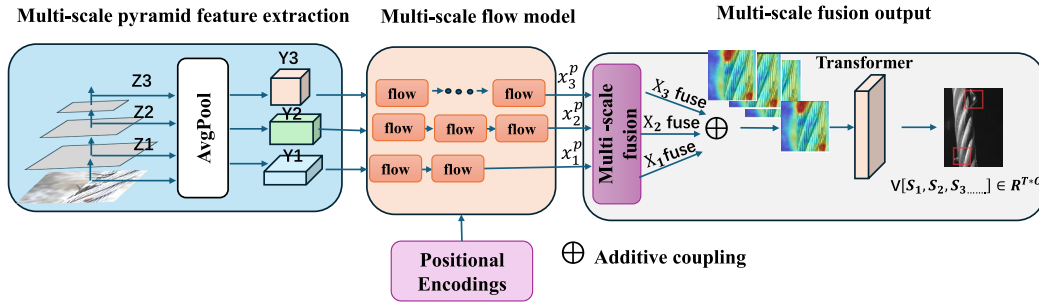


Figure 4: Defect detection network structure based on normalizing flows.

The multi-scale flow model is a bidirectional invertible generative network that maps the data distribution $p_y(y)$ to a tractable base distribution $p_x(x)$. Inspired by FastFlow and MSFlow [34], the model establishes the mapping between samples x and latent variables z :

$$x = f(z) = f_k \circ f_{k-1} \circ \dots \circ f_1(z) \quad (13)$$

$$z = f^{-1}(x) = f_1^{-1} \circ f_2^{-1} \circ \dots \circ f_k^{-1}(x) \quad (14)$$

Here, K is the number of flow modules, and $\circ$ denotes successive transformations. Fig. 5 illustrates the affine coupling operation inside each flow module used in these transformations.

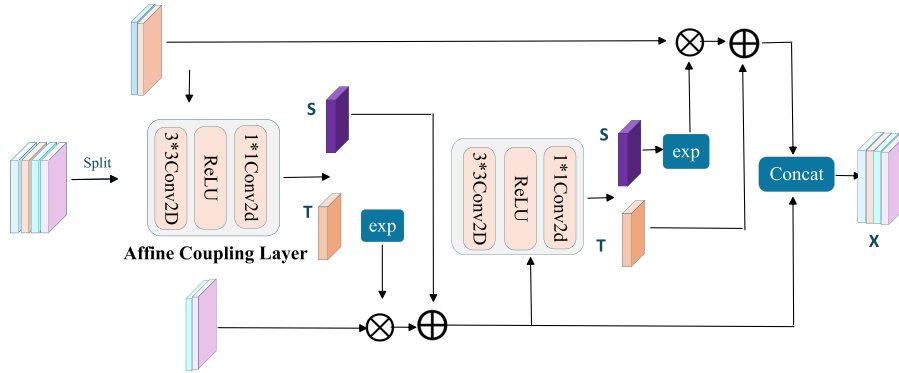


Figure 5: Flow module network structure.

Each flow module consists of two affine coupling layers. The channel partition, affine transformation, and recombination shown in Fig. 5 correspond to $\text{split}(\cdot)$, $s(\cdot)$, $t(\cdot)$, and $\text{concat}(\cdot)$ in Eq. (15).

$$\begin{aligned} y_a, y_b &= \text{split}(y), \\ y'_a &= y_a, \\ y'_b &= s(y_a) \odot y_b + t(y_a), \\ y' &= \text{concat}(y'_a, y'_b), \\ s, t(\cdot) &= \text{Conv}_{3 \times 3} \circ \text{ReLU} \circ \text{LN} \circ \text{Conv}_{1 \times 1}. \end{aligned} \quad (15)$$

Here, $s(y_a)$ and $t(y_a)$ are neural network outputs, while $\text{split}(\cdot)$ and $\text{concat}(\cdot)$ perform channel partitioning and concatenation. The $s(\cdot)$ and $t(\cdot)$ subnetworks use alternating 3×3 and 1×1 convolutions, with positional encoding added to capture location dependent information. The multi-scale outputs $\{x_i^p\}_{i=1}^3$ are fused into $\{x_i^{fuse}\}_{i=1}^3$.

Fig. 6 shows the implementation of the fusion mapping f_{fuse} in Eq. (18). The branch outputs $\{x_i^p\}_{i=1}^3$ are resized to the smallest resolution, concatenated, and processed by 3×3 and 1×1 convolutions. The fused representation is then split and upsampled to the original resolutions, and residual connections produce $\{x_i^{fuse}\}_{i=1}^3$.

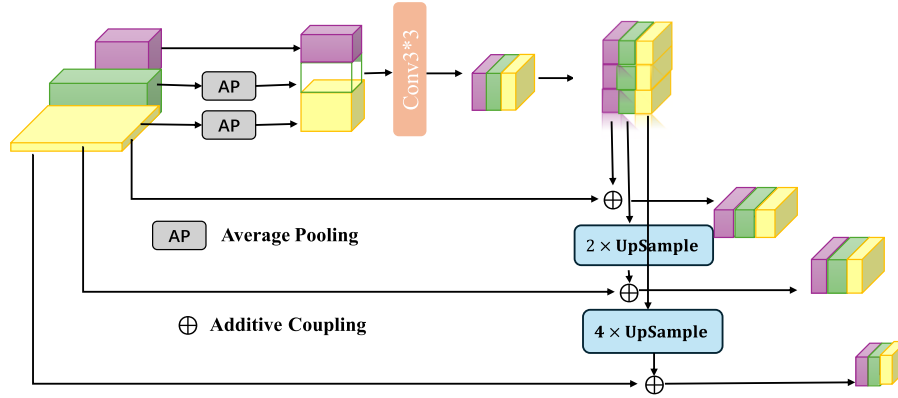


Figure 6: Multi-scale fusion output network structure.

By the change of variables theorem, $p_x(x) = p_z(z)|\det(J_{f^{-1}}(x))|$, where $J_{f^{-1}}(x) = \frac{\partial f^{-1}(x)}{\partial x}$. Maximum likelihood estimation is therefore equivalent to minimizing the negative log likelihood:

$$\max p_x(x) \Leftrightarrow \min -\log p_x(x) \Leftrightarrow \min (-\log p_z(z) - \log|\det J_{f^{-1}}(x)|) \quad (16)$$

With a multivariate standard Gaussian base distribution, the objective becomes:

$$\max p_x(x) \Leftrightarrow \min \left(\frac{1}{2} \|z\|_2^2 - \log|\det J_{f^{-1}}(x)| \right) \quad (17)$$

For multi-scale features, ADFN maximizes the likelihood of $p_y(y)$ through flow branches $\{f_i^p\}_{i=1}^3$ and fusion module f_{fuse} :

$$x_i^p = f_i^p(y_i), \quad (x_1^{fuse}, x_2^{fuse}, x_3^{fuse}) = f_{fuse}(x_1^p, x_2^p, x_3^p) \quad (18)$$

The chain rule jointly evaluates $|\det J_{f_i^{p-1}}|$ and $|\det J_{f_{fuse}^{-1}}|$, yielding the loss:

$$\begin{aligned} L = -\log p_y(y) &= \sum_{i=1}^3 \frac{1}{2} \|x_i^{fuse}\|_2^2 - \log|\det J_{f^{-1}}| \\ &= \sum_{i=1}^3 \frac{1}{2} \|x_i^{fuse}\|_2^2 - \left[\sum_{i=1}^3 \log|\det J_{f_i^{p-1}}| + \log|\det J_{f_{fuse}^{-1}}| \right]. \end{aligned} \quad (19)$$

During training, defect-free frames are used to model the normal feature manifold. For a test sample, the likelihood at pixel location (j, k) is:

$$\log \hat{p}_y(y_{ijk}) = -\frac{1}{2} \|z_{ijk}\|_2^2 + \log |\det J_{f^{-1}}(y_{ijk})| \quad (20)$$

Here, $\log |\det J_{f^{-1}}(y_{ijk})|$ is the Jacobian determinant at (j, k) . To reduce local noise, anomaly levels are computed from feature map NLL, which reduces sensitivity to isolated pixel likelihoods. Multi-scale likelihood maps are bilinearly upsampled and summed into P_{add} . The pixel-level score is:

$$P_{\text{add}} = \sum_{i=1}^3 e^{\log P_i} = \sum_{i=1}^3 P_i, \quad (21)$$

$$S_{\text{loc}} = \max(P_{\text{add}}) - P_{\text{add}}$$

Here, larger S_{loc} indicates higher defect likelihood.

After spatial detection, local defect scores are aggregated into frame-level scores $S_{\text{global}} = [s_1, s_2, \dots, s_T]$ for Transformer temporal modeling. The Transformer uses this score sequence as input, leaving image features unchanged, and the implemented sliding window refinement is summarized as:

$$\begin{aligned} \mathbf{s}^{(m)} &= [s_m, s_{m+1}, \dots, s_{m+L-1}], \quad L = 16, \\ \tilde{\mathbf{s}}^{(m)} &= \text{clip}(\mathbf{s}^{(m)} + \Delta_\theta(\mathbf{s}^{(m)}, M_c), 0, 1), \\ \hat{s}_t &= \text{mean}_{m \in \mathcal{W}_t}(\tilde{s}_{t-m+1}^{(m)}). \end{aligned} \quad (22)$$

Here, $\Delta_\theta(\cdot)$ denotes the bounded residual correction predicted by the Transformer with positional encoding and causal mask M_c , and $\mathcal{W}_t$ is the set of windows covering frame t . This formulation refines frame-level score trajectories and suppresses isolated peaks caused by illumination, dust, or vibration.

3 Experiments

3.1 Experimental Settings and Dataset

Experiments were conducted on a Linux Ubuntu 4.15.0 training server with an Intel Xeon Gold 6248R central processing unit (CPU) and NVIDIA Tesla P100 PCIe graphics processing unit (GPU). Inference was evaluated on a Windows 11 platform with an Intel Core i7-10700K CPU and NVIDIA RTX 3060 GPU. The ADFN spatial detector used an initial learning rate of 1×10^{-3} , batch size 4, weight decay 5×10^{-4} , and 100 training epochs. A pretrained ResNet50 retained the first three convolutional stages; n was selected from 1 to 10, with $\lambda = 100$ and $L = 3$.

The temporal Transformer uses windows of 16 anomaly scores with stride 1 for causal inference over frames. Scalar scores are projected to a 64-dimensional embedding and processed by two encoder layers, each with four attention heads and a 128-dimensional feed forward subnetwork. Dropout is 0.1, and sinusoidal positional encoding with causal masking prevents access to future scores. This compact temporal module contains 0.069313 M trainable parameters.

The model is trained only on defect-free anomaly score sequences using AdamW and Smooth L1 loss. Training uses batch size 64, learning rate 1×10^{-4} , weight decay 1×10^{-5} , at most 100 epochs, patience 15, and random seed 9826. Gaussian noise (standard deviation 0.02) and transient positive spikes (probability 0.05, width 1–3 frames, amplitude 0.2–0.8) are added to the input scores. The unperturbed normal score sequence serves as the target, so no defective videos enter temporal model training.

The complete training process required approximately 16–20 h on the NVIDIA Tesla P100 PCIe GPU, including MEPem pretraining, ADFN training, and temporal score modeling. Peak GPU memory usage during the main combined training stage was approximately 11–12 GB. Because the temporal Transformer operates on scalar score sequences, its training overhead is limited relative to MEPem and ADFN.

For evaluation with thresholds, the decision threshold was set to the 99th percentile of anomaly scores from defect-free validation frames and then applied unchanged to the test set. Each baseline used the same validation percentile rule with its own validation scores to account for the score scale of each method. A threshold sensitivity check using the 95th, 97th, 98th, 99th, and 99.5th validation percentiles yielded FAR/recall pairs of 21.01%/100.0%, 15.94%/100.0%, 10.97%/100.0%, 8.40%/100.0%, and 8.40%/99.6%, respectively, on the test set. The 99th percentile was used as a conservative operating point for the evaluated data; under stronger distribution shifts beyond the collected conditions, such as severe oil mist or water mist, recalibration for each site using defect-free calibration videos may be required.

The dataset was collected using 304 stainless steel wire ropes complying with GB/T standards. Four 7×19 ropes with a 24 mm diameter match typical mine hoist specifications. Data were captured by a Hikvision MV-CS050-10GC camera with a KF3528M-12MP lens for close range imaging and a Hikvision DS-2ZCN2006 network camera for streaming and global monitoring. Adjustable light-emitting diode (LED) ring lights simulate strong light, low light, and flicker, while a rail system with speeds of 0.5–2 m/s and mechanical vibration reproduces high speed blur.

Referring to International Organization for Standardization (ISO) 4309 inspection criteria, this study focuses on safety related broken wire, wear, and corrosion defects associated with later discard risk assessment. The task is defined as video defect screening to support discard risk assessment. The dataset contains 800 curated 5–10 s videos: 500 defect-free videos and 300 defect sample videos. The defect sample subset includes 100 broken wire, 120 wear, and 80 corrosion videos; 45, 65, 50, 25, and 115 of them were recorded under normal illumination, strong light, low light, flicker, and motion blur, respectively. Following the unsupervised anomaly detection protocol in which training uses only defect-free videos, the 500 defect-free videos are split into training, validation, and test subsets at 7:2:1. The 300 defect sample videos are excluded from training; 100 are retained only for validation, and 200 are used for final testing. Threshold selection uses only defect-free validation frames, as described above, and [Table 1](#) summarizes the allocation of videos. The 350 defect-free training videos cover the predefined combinations of motion modes and environmental conditions in the collected dataset, including illumination variation, rope speed changes, vibration, and motion blur. The defect-free validation and test videos are independent recordings within the same condition range and do not overlap with the training videos. Some defective test videos contain combined degradation factors formed from the listed imaging conditions and defect categories.

Table 1: Dataset allocation for training with only defect-free videos.

Subset	Defect-Free Videos	Defect Sample Videos	Total Videos	Usage
Training	350	0	350	Normal distribution learning
Validation	100	100	200	Validation; threshold from defect-free videos
Testing	50	200	250	Final evaluation
Total	500	300	800	–

All videos were acquired at 25 frames per second (FPS) with an original resolution of 900×900 pixels, and frames were resized to 512×512 pixels before model input. Each 5–10 s clip contains 125–250 frames, giving approximately 100,000–200,000 frames before resizing.

3.2 Comparative Experiment Analysis

The proposed method is compared with representative industrial anomaly detection baselines: DifferNet [21], CS-Flow [22], CFlow-AD [23], FastFlow [24], PatchCore [25], PyramidFlow [28], and RareCLIP [35]. These baselines cover models based on normalizing flows, a feature embedding method with a memory bank, and a zero shot reference method. Evaluation uses frame-level area under the receiver operating characteristic curve (AUC-ROC), parameter count, false alarm rate (FAR), and FPS. For PatchCore, the image-level anomaly score is used as the frame-level score, and its pixel-level heatmap is not used in this protocol. All methods are evaluated on the same dataset split and inference platform, with the same threshold selection rule for metrics that depend on thresholds (Table 2).

Table 2: Comparative experimental results.

Method	AUC-ROC (%)	Params (M)	FAR (%)	FPS
DifferNet	88.9	60.3	12.0	22.4
CFlow-AD	90.8	81.6	11.2	23.7
CS-Flow	91.7	80.5	9.8	23.9
FastFlow	90.4	66.9	10.4	24.8
PatchCore	94.4	68.9*	8.6	3.1
PyramidFlow	92.9	54.3	9.8	21.2
RareCLIP	94.1	427.9	8.8	10.7
Ours	93.4	60.1	8.4	25.1

Note: *For PatchCore, Params refers to the backbone network; the memory bank storage is 278.4 MB in our implementation.

Table 2 shows that the proposed method reaches 93.4% AUC-ROC, exceeding the compared models based on normalizing flows by 0.5–4.5 percentage points. Compared with FastFlow, it improves AUC-ROC by 3.0 percentage points and reduces FAR from 10.4% to 8.4% while maintaining real time speed. CFlow-AD and CS-Flow require more than 80 M parameters, and PyramidFlow lacks temporal constraints for continuous videos.

PatchCore achieves the highest AUC-ROC (94.4%) but has a slightly higher FAR (8.6%) and lower speed (3.1 FPS) because nearest neighbor retrieval over the memory bank is required. Its frame-level protocol also does not explicitly suppress transient responses from video degradation. RareCLIP achieves 94.1% AUC-ROC, but its 427.9 M parameters and 10.7 FPS constrain online use in the tested setting. The proposed method reaches 93.4% AUC-ROC and 8.4% FAR at 25.1 FPS, providing a practical tradeoff for online wire rope monitoring.

Module visualizations are reported before the final localization comparison. The examples in Figs. 7–9 are selected from different illumination and motion conditions to illustrate rope surface detail recovery across representative scenes. Fig. 7 shows VFE enhancement under low illumination.

Fig. 8 shows that SFID improves the clarity of rope textures and defect boundaries.

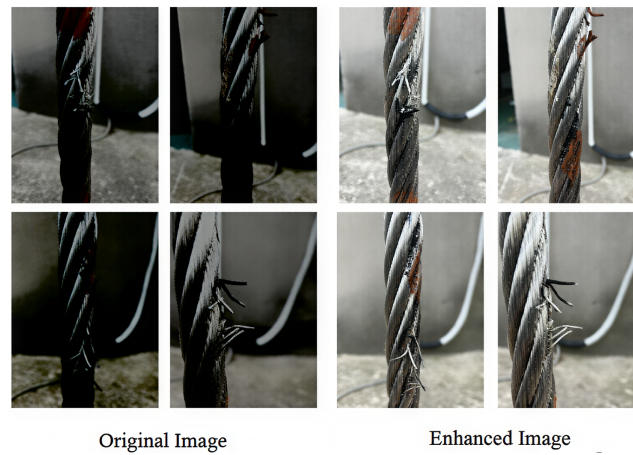


Figure 7: Comparison of VFE enhancement results.



Figure 8: Results after SFID processing.

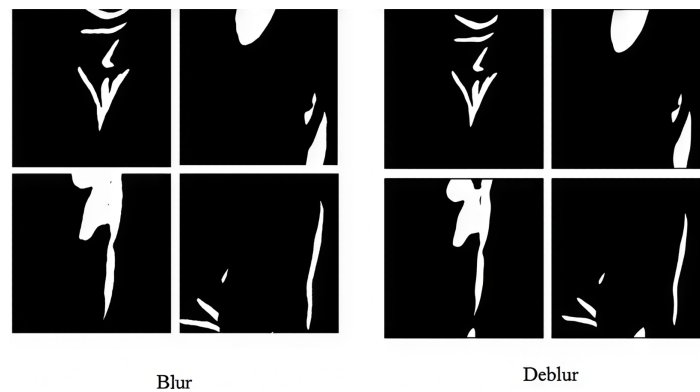


Figure 9: Defect region masks.

This experiment examines whether SFID deblurring alters the visible defect region area that provides spatial evidence for subsequent anomaly detection. Matched ROIs before and after SFID processing were manually annotated; [Fig. 9](#) shows representative defect region masks, and [Table 3](#) reports the corresponding area changes.

Table 3: Defect region mask area changes before and after SFID.

Defect Type	Before SFID Area (%)	After SFID Area (%)	Abs. Change (pp)	Rel. Change (%)
Broken wire	2.4012	2.3952	0.0060	0.25
Corrosion	4.6123	4.6069	0.0054	0.12
Wear	5.4012	5.3722	0.0290	0.54

Fig. 10 compares localization under degraded illumination and different defect scales. The baselines either miss subtle broken wires, produce expanded responses, show discontinuous contours, or generate dispersed local activations. The proposed method yields more concentrated defect responses while maintaining real time inference speed and a low frame-level FAR.

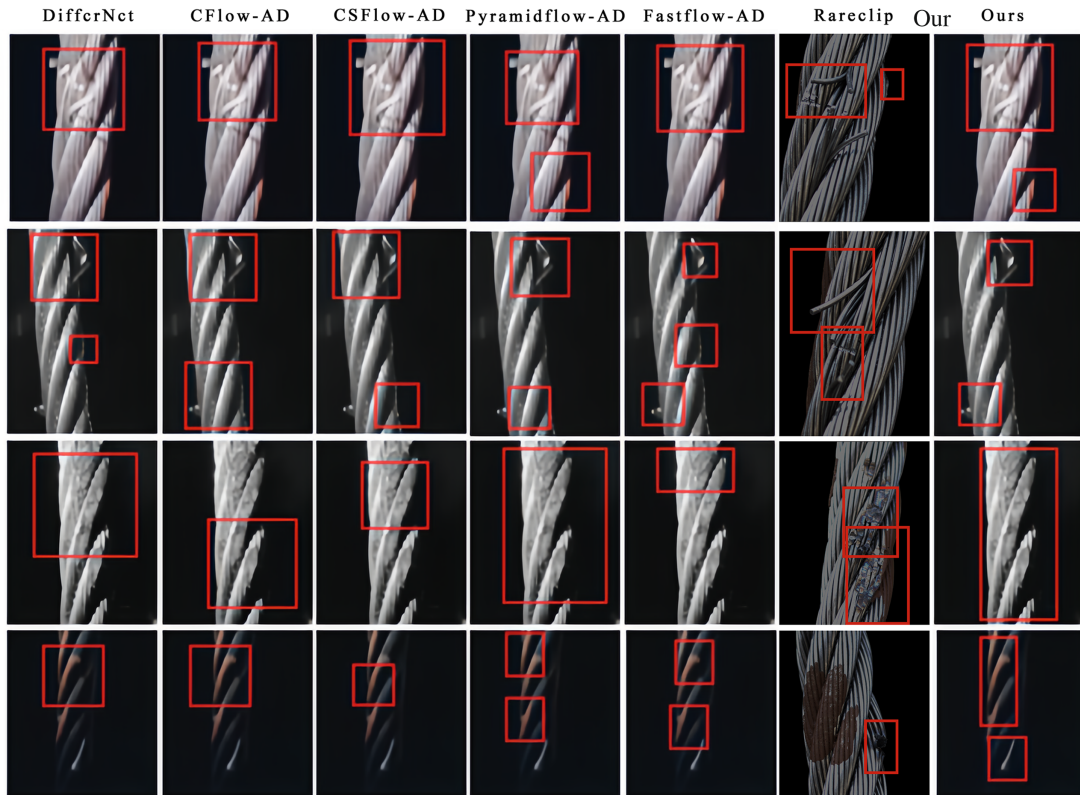


Figure 10: Visual comparison of defect localization results obtained by normalizing flow methods, RareCLIP, and the proposed method.

3.3 Ablation Experiment Analysis

Ablation experiments were conducted under the same hardware and hyperparameter settings for input degradation compensation (VFE and SFID), multi-scale spatial anomaly modeling (MSFE, MSFM, and MSF), and temporal false alarm suppression (Transformer). The results are summarized in [Table 4](#).

[Table 4](#) shows that directly applying the flow model yields 50.2% AUC-ROC and 58.5% FAR, indicating the difficulty of modeling degraded frames without enhancement or multi-scale representation. VFE raises AUC-ROC to 62.3% and lowers FAR to 40.2%, and adding SFID further improves AUC-ROC to 73.4% and FAR to 27.4%.

Table 4: Ablation experimental results.

VFE	SFID	MSFE	MSFM	MSF	AUC-ROC (%)	Params (M)	FAR (%)	FPS
–	–	–	–	–	50.2	–	58.5	30.8
✓	–	–	–	–	62.3	23.5	40.2	29.6
✓	✓	–	–	–	73.4	26.4	27.4	28.8
✓	✓	✓	–	–	80.2	30.2	15.2	27.3
✓	✓	✓	✓	–	88.7	50.3	12.4	26.5
✓	✓	✓	✓	✓	93.4	60.1	8.4	25.1

Note: ✓ indicates that the module is included; – indicates that the module is not used or the value is not applicable.

The MEPem configuration was further evaluated by replacing VFE and SFID with representative alternatives, Zero-DCE [36] and DeblurGAN [37], while keeping ADFN and the Transformer unchanged (Table 5).

Table 5: Replacement comparison of MEPem configurations.

MEPEM Configuration	AUC-ROC (%)	FAR (%)	FPS
Zero-DCE + SFID	91.6	9.9	23.0
VFE + DeblurGAN	91.8	11.1	21.3
Zero-DCE + DeblurGAN	89.6	12.8	17.9
VFE + SFID (Ours)	93.4	8.4	25.1

Table 5 shows that VFE + SFID obtains the highest AUC-ROC and FPS and the lowest FAR among the evaluated MEPem configurations, supporting the MEPem design for downstream wire rope anomaly detection.

For ADFN, adding MSFE, MSFM, and MSF raises AUC-ROC from 73.4% to 93.4% and lowers FAR from 27.4% to 8.4%. Although the full configuration has 60.1 M parameters and reduces speed from 30.8 to 25.1 FPS, the final speed satisfies the real time requirement of the 25 FPS mine hoist wire rope inspection setting.

Temporal variants were evaluated with fixed MEPem and ADFN because they refine frame-level anomaly score sequences. The fluctuation index is defined as $J = (T - 1)^{-1} \sum_{t=2}^T |s_t - s_{t-1}|$, where s_t is the frame-level anomaly score.

Table 6 shows small AUC-ROC differences across temporal variants, while FAR and fluctuation J better reflect temporal stability. iTransformer (Ours) obtains the lowest FAR and J , with 0.6 percentage points higher AUC-ROC than long short-term memory (LSTM) and gated recurrent unit (GRU). Although the recurrent baselines are slightly faster, all variants remain real time. These results indicate that iTransformer mainly improves score stability and false alarm control under the same complete pipeline setting.

The temporal window length was varied while the remaining pipeline was fixed (Table 7).

The 16-frame setting provides the highest AUC-ROC and lowest FAR and was selected as the default temporal window length.

Table 6: Temporal module replacement comparison under the complete pipeline.

Temporal Strategy	AUC-ROC (%)	FAR (%) ↓	Fluctuation J ↓	FPS
Frame-level only	93.1	11.60	0.08600	26.3
Temporal Smoothing	93.2	9.80	0.06100	25.9
LSTM	92.8	9.62	0.04643	28.0
GRU	92.8	9.21	0.04842	28.2
iTransformer (Ours)	93.4	8.42	0.04346	25.1

Note: ↓ indicates that lower values are preferred.

Table 7: Sensitivity analysis of transformer window length.

Window Length	AUC-ROC (%)	FAR (%) ↓	Fluctuation J ↓	FPS
8	93.1	8.75	0.04340	25.6
16	93.4	8.42	0.04346	25.1
32	92.9	11.47	0.04532	24.0

Note: ↓ indicates that lower values are preferred.

ADFN was also evaluated on the MVTec Anomaly Detection (MVTec AD) cable category [38] using defect-free cable images for training. This experiment provides a supplementary check of ADFN spatial flow modeling on cable-like objects (Table 8).

Table 8: Supplementary validation on the MVTec AD cable category.

Dataset	Category	Det. AUC-ROC (%)	Loc. AUC-ROC (%)
MVTec AD	Cable	96.31	95.29

ADFN achieved 96.31% Det. AUC-ROC and 95.29% Loc. AUC-ROC on this category. MVTec AD Cable differs from mine hoist wire ropes in material, texture, diameter, strand structure, and imaging conditions; therefore, the claim is limited to preliminary support for ADFN spatial feature extraction. Broader validation of the complete video framework across other MVTec AD categories and field cable inspection scenarios remains future work.

The ablation results show that VFE and SFID address input degradation, MSFE/MSFM/MSF strengthen multi-scale anomaly representation, and the Transformer reduces score fluctuations and false alarms while preserving real time inference under the evaluated mine hoist conditions.

4 Conclusion

This study presents a feature embedding method based on normalizing flows for mine hoist wire rope video defect detection with scarce defect samples and degraded imaging, with training performed only on defect-free videos. MEPEM compensates for low illumination and motion blur, ADFN models multi-scale spatial likelihoods, and the causal Transformer at the score level stabilizes frame-level anomaly scores. The method achieves 93.4% frame-level AUC-ROC with 60.1 M parameters, reduces FAR to 8.4%, and reaches 25.1 FPS, supporting wire rope video monitoring under degraded underground imaging. The current evaluation is limited to the collected videos and considered defect categories; future work will examine longer

underground hoist recordings, additional rope specifications, and more complex conditions such as oil mist and water mist before broader deployment in mine sites.

Acknowledgement: Not applicable.

Funding Statement: This research was funded by the Science and Technology Innovation 2030—Major Project of New Generation Artificial Intelligence, grant number 2022ZD0119005, the 2025 Shaanxi Provincial Natural Science Basic Research Program, grant number 2025JC-YBMS-754, and the National Key Laboratory of Spatial Intelligent Control Technology (HTKJ2025SY502002).

Author Contributions: Tian Ma contributed to the conceptualization of the study, methodology design, investigation, data curation, visualization, and preparation of the original manuscript. Tong An contributed to the methodological development, software implementation, experimental validation, formal analysis, and manuscript revision. Jiahao Wang contributed to data acquisition, experimental implementation, visualization, and validation. Ruping Zhang supervised the study, administered the project, provided funding support, and contributed to manuscript review and editing. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The MVTec AD cable dataset used for supplementary validation is publicly available and is cited in this article. The custom mine hoist wire rope video dataset generated and analyzed in this study is not publicly available because the videos were collected at a real industrial company site and contain proprietary operating scenes, equipment information, and safety-sensitive production conditions. Aggregated dataset statistics, splits, and evaluation results supporting the findings are included in this article. Access to the custom dataset may be considered by the corresponding author upon reasonable request, subject to permission from the data-providing company and applicable confidentiality agreements.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yu B, Xie C. Method for detecting industrial defects in intelligent manufacturing using deep learning. *Comput Mater Contin.* 2024;78(1):1329–43. doi:10.32604/cmc.2023.046248.
2. Luo T, Zhou Y, Shi D, Yun Q, Wang S, Zhang J, et al. A lightweight defect detection transformer for printed circuit boards combining image feature augmentation and refined cross-scale feature fusion. *Eng Appl Artif Intell.* 2025;155:111128. doi:10.1016/j.engappai.2025.111128.
3. Noé DG, Lin K, Chuang C, Cheng Y. Applying deep learning to defect detection in steel manufacturing. *Comput Mater Contin.* 2026;88(1):7. doi:10.32604/cmc.2026.077838.
4. Tulbure AA, Tulbure AA, Dulf EH. A review on modern defect detection models using DCNNs—deep convolutional neural networks. *J Adv Res.* 2022;35(2):33–48. doi:10.1016/j.jare.2021.03.015.
5. Hao R, Lu B, Cheng Y, Li X, Huang B. A steel surface defect inspection approach towards smart industrial monitoring. *J Intell Manuf.* 2020;32(7):1833–43. doi:10.1007/s10845-020-01670-2.
6. Zhong Z, Wang H, Zhou C, Kang H, Zhao J, Hu Y. Structure-preserving surface defect detection for flexible integrated circuit substrates. *Image Vis Comput.* 2026;171(2):106007. doi:10.1016/j.imavis.2026.106007.
7. Liu R, Shen Z, Zhao L, Ren Y, Zhang L, Li L, et al. A lightweight ultrasound welding surface defect detection network based robust feature downsampling. *Appl Soft Comput.* 2026;192(3):114736. doi:10.1016/j.asoc.2026.114736.
8. Wang G, Wang S, Tao R, Li N, Luo Z. Research on visual recognition technology for appearance defects of steel wire rope in mine hoist. *J Mine Autom.* 2024;50(5):28–35. (In Chinese).
9. Gao Y, Li X, Wang XV, Wang L, Gao L. A review on recent advances in vision-based defect recognition towards industrial intelligence. *J Manuf Syst.* 2021;62:753–66. doi:10.1016/j.jmsy.2021.05.008.

10. Shi L, Tan J, Xue S, Deng J. Inspection method of rope arrangement in the ultra-deep mine hoist based on optical projection and machine vision. *Sensors*. 2021;21(5):1769–90. doi:10.3390/s21051769.
11. Liang B, Song X, Gao J, Liu T. Defect detection method of steel wire rope based on YOLO algorithm. *Nondestruct Test*. 2023;47(6):1–8. (In Chinese). doi:10.26855/acc.2025.04.001.
12. Jiang H, Dong Z, He Z. Surface defect detection of wire rope based on feature fusion and IWOA-SVM. *J Taiyuan Univ Sci Technol*. 2023;44(5):434–46. (In Chinese).
13. Aiger D, Talbot H. The phase only transform for unsupervised surface defect detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; 2010 Jun 13–18; San Francisco, CA, USA. p. 295–302.
14. Chen S, Li Y, Lin W, Lai J, Chao J. Unsupervised anomaly detection in industrial images based on autoencoder feature discrepancy. *Comput Eng Des*. 2025;46(11):3199–206. (In Chinese).
15. Zavrtnik V, Kristan M, Skocaj D. A discriminatively trained reconstruction embedding for surface anomaly detection. *arXiv:2108.07610*. 2021.
16. Zhang X, Xu M, Zhou X. RealNet: a feature selection network with realistic synthetic anomaly for anomaly detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2024 Jun 17–21; Seattle, WA, USA. p. 16699–708.
17. Batzner K, Heckler L, Koenig R. EfficientAD: accurate visual anomaly detection at millisecond-level latencies. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*; 2024 Jan 3–8; Waikoloa, HI, USA, pp. 128–38.
18. Rippel O, Mertens P, Merhof D. Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In: *Proceedings of the 25th International Conference on Pattern Recognition*; 2021 Jan 10–15; Milan, Italy. p. 6726–33.
19. Zhang K, Wang B, Kuo CCJ. PEDENet: image anomaly localization via patch embedding and density estimation. *Pattern Recognit Lett*. 2022;153:144–50.
20. Wan Q, Gao L, Li X, Wen L, Feng L. Unsupervised image anomaly detection and segmentation based on pretrained feature mapping. *IEEE Trans Ind Inform*. 2022;19(3):2330–9. doi:10.1109/tii.2022.3182385.
21. Rudolph M, Wandt B, Rosenhahn B. Same same but DifferNet: semi-supervised defect detection with normalizing flows. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*; 2021 Jan 5–9; Waikoloa, HI, USA. p. 1907–16.
22. Rudolph M, Wehrbein T, Rosenhahn B, Wandt B. Fully convolutional cross-scale-flows for image-based defect detection. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*; 2022 Jan 3–8; Waikoloa, HI, USA. p. 1088–97.
23. Gudovskiy D, Ishizaka S, Kozuka K. CFlow-AD: real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*; 2022 Jan 4–8; Waikoloa, HI, USA. p. 98–107.
24. Yu J, Zheng Y, Wang X, Li W, Wu Y, Zhao R, et al. FastFlow: unsupervised anomaly detection and localization via 2D normalizing flows. *arXiv:2111.07677*. 2021.
25. Roth K, Pemula L, Zepeda J, Scholkopf B, Brox T, Gehler P. Towards total recall in industrial anomaly detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2022 Jun 19–24; New Orleans, LA, USA. p. 14318–28.
26. Yan R, Zhang F, Huang M, Liu W, Hu D, Li J, et al. CAINNFlow: convolutional block attention modules and invertible neural networks flow for anomaly detection and localization tasks. *arXiv:2206.01992*. 2022.
27. Kim Y, Jang H, Lee DK, Choi HJ. AltUB: alternating training method to update base distribution of normalizing flow for anomaly detection. *arXiv:2210.14913*. 2022.
28. Lei J, Hu X, Wang Y, Liu D. PyramidFlow: high-resolution defect contrastive localization using pyramid normalizing flow. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2023 Jun 18–22; Vancouver, BC, Canada. p. 14143–52.
29. Liu J, Yan Y, Li J, Zhao W, Chu P, Sheng X, et al. IPAD: industrial process anomaly detection dataset. *IEEE Trans Circuits Syst Video Technol*. 2024;35(1):380–93.

30. Shukla V, Shukla A, Surya Prakash SK, Shukla S. A systematic survey: role of deep learning-based image anomaly detection in industrial inspection contexts. *Front Robot AI*. 2025;12:1554196.
31. Dilek E, Dener M. An overview of transformers for video anomaly detection. *Neural Comput Appl*. 2025;37(22):17825–57. doi:10.1007/s00521-025-11218-1.
32. Wu P, Pan C, Yan Y, Pang G, Yan Q, Wang P, et al. Deep learning for video anomaly detection: a review. *IEEE Trans Neural Netw Learn Syst*. 2026;37(7):3010–30.
33. Ma T, Fu C, Yang J, Zhang J, Shang C. RF-Net: unsupervised low-light image enhancement based on Retinex and exposure fusion. *Comput Mater Contin*. 2023;77(1):1103–22. doi:10.32604/cmc.2023.042416.
34. Zhou Y, Xu X, Song J, Shen F, Shen HT. MSFlow: multiscale flow-based framework for unsupervised anomaly detection. *IEEE Trans Neural Netw Learn Syst*. 2025;36(2):2437–50.
35. He J, Cao M, Peng S, Xie Q. RareCLIP: rarity-aware online zero-shot industrial anomaly detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*; 2025 Oct 19–23; Honolulu, HI, USA. p. 24478–87.
36. Guo C, Li C, Guo J, Loy CC, Hou J, Kwong S, et al. Zero-reference deep curve estimation for low-light image enhancement. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2020 Jun 13–19; Seattle, WA, USA. p. 1780–9.
37. Kupyn O, Budzan V, Mykhailych M, Mishkin D, Matas J. DeblurGAN: blind motion deblurring using conditional adversarial networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; 2018 Jun 18–22; Salt Lake City, UT, USA. p. 8183–92.
38. Bergmann P, Fauser M, Sattlegger D, Steger C. MVTec AD—a comprehensive real-world dataset for unsupervised anomaly detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2019 Jun 16–20; Long Beach, CA, USA. p. 9584–92.