Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.080767

ARTICLE

LAH-Net: A Low-Light Aware Hybrid Network for Robotic Manipulation

Yingying Yu[1,2,*,#], **Jun Yuan**[3,#] **and Tong Liu**[1,2]

[1]Beijing Academy of Science and Technology, Beijing, China
[2]Beijing Key Laboratory of Traffic Data Mining and Embodied Intelligence, Beijing, China
[3]Independent Researcher, Beijing, China

*Corresponding Author: Yingying Yu. Email: yuyingying@bjast.ac.cn

[#]These authors contributed equally to this work

Received: 18 February 2026; Accepted: 07 April 2026; Published: 08 May 2026

ABSTRACT: Accurate grasp detection is fundamental to successful robotic manipulation. Existing methods achieve reliable performance under good light conditions. However, their performance in low-light environments suffers from severe degradation due to the diminishing discriminative ability of visual features. In this paper, a novel low-light aware hybrid network LAH-Net is proposed. It comprises an alternating transformer-CNN module (ATCM) between the encoder and decoder, and a knowledge distillation-guided low-light enhancement module (KDLEM) before the encoder, which is activated by an illumination gate under low-light conditions. To generate highly robust and synergistic features, the ATCM module facilitates the iterative exchange between the local representations from CNNs and the global contexts modeled by transformers. Additionally, a transformer-to-CNN adapter and a CNN-to-transformer adapter are designed for bidirectional feature alignment. Meanwhile, the KDLEM module employs a teacher-student framework where the simplified student network distills knowledge from the powerful teacher to enhance low-light adaptability and maintain computational efficiency. Moreover, we introduce a real-world low-light grasp detection dataset (RLGD) for algorithm evaluation, which contains over 70 objects captured under four distinct low-light conditions. Our method achieves 99.4% and 98.8% accuracy on the Cornell dataset and 96.05% accuracy on the Jacquard V2 dataset. It also attains an accuracy of 96.5% on the RLGD dataset and demonstrates strong generalization across various low-light intensity levels. The real-world experiments in low-light scenarios validate the effectiveness of the proposed method.

KEYWORDS: Robotic manipulation; grasp detection; low-light enhancement; alternating transformer-CNN; grasping dataset

1 Introduction

Robotic grasp detection plays a pivotal role in industrial automation, logistics, and domestic services. The ability to precisely identify graspable regions directly impacts operational efficiency and safety, particularly in unstructured environments [1–3]. Existing methods perform well under good lighting conditions [4,5], but their performance degrades significantly in low-light environments, such as nighttime warehouses and underground mining, where visual features become less reliable.

We believe that existing methods have two key limitations in low-light grasping [6,7]. First, there is insufficient interaction between CNNs and transformers. Low-light images often exhibit spatially non-uniform illumination degradation, which requires first capturing global illumination priors, then refining local details, and finally performing global grasp reasoning. A one-time feature fusion strategy cannot fully

satisfy this requirement. Second, there is a mismatch between the objectives of image enhancement and grasp detection. Existing methods either treat enhancement as a preprocessing step or rely on synthetic data generation, lacking task-driven adaptive enhancement.

Unlike traditional methods, which mainly rely on handcrafted features to align objects with predefined grasp configurations, recent grasp detection methods are based on CNNs [8–10]. CNN-based methods [11–13] are widely used as it generates pixel-wise grasp detection results for each pixel position. Such dense prediction paradigm provides comprehensive candidate options for grasp detection. But their limited receptive field makes it difficult to capture global illumination distribution. With the successful application of transformer in natural language processing and computer vision [14], researchers have adopted transformers for grasp detection. Transformers [15] can improve detection accuracy by leveraging their ability to capture long-range dependencies. However, they have high computational cost and are less effective at extracting local details. Therefore, researchers have attempted to combine transformers with CNNs to take advantage of both approaches. They typically embed transformer modules into the CNN-based encoder or decoder in a sequential [16,17] or parallel way [18]. However, these methods perform feature fusion only once and cannot achieve iterative interaction between global and local information, which is especially important in low-light scenarios. In addition to grasp detection, recent studies have also emphasized the importance of real-time visual tracking and control accuracy for the end effector [3]. This highlights the necessity of achieving stable visual perception throughout the entire mechanical pipeline.

One possible approach for low-light environments is to integrate a camera with other modalities including infrared [19] and radar [20], which compensates the possible visual degradation but with engineering complexity. Depth camera-based solutions are more popular. However, their performance under low-light conditions often degrades due to poor image quality. To address this, a common manner is to directly incorporate a pre-trained image enhancement module at the front end of the workflow [21]. This pre-processing step aims to improve visual characteristics such as brightness and noise reduction [22–24]. Nevertheless, it may weaken geometric features essential for grasp detection, with critical structural information degraded by excessive smoothing or distorting. Moreover, errors from the enhancement process, including artifacts or over-enhancement, can propagate into and adversely affect downstream grasp detection performance. Other approaches explore the use of synthetic data generated under low-light conditions to improve network adaptability [25]. To avoid the disconnection between image enhancement and grasp detection, joint optimization strategies are adopted with the generation of low-light training data [26,27]. However, their performance may degrade significantly when applied to scenarios that are unseen or with different low-light levels. Moreover, models trained on synthetic data often perform poorly in practice because of real-world complexity. These methods have limited generalization ability and lack a task-driven adaptive enhancement mechanism.

To address the above issues, a novel low-light aware hybrid network LAH-Net is proposed. To tackle the first limitation, an alternating transformer-CNN module (ATCM) is designed, which adopts a transformer-CNN-transformer sequence to achieve iterative collaboration of global and local features. To address the second limitation, a knowledge distillation-guided low-light enhancement module (KDLEM) is proposed. This module integrates enhancement learning with grasp detection through a teacher-student framework and employs illumination gating for adaptive activation. The main contributions of this work are summarized as follows:

(1) A novel low-light aware hybrid network LAH-Net is proposed in an encoder-decoder architecture. It incorporates an alternating transformer-CNN module (ATCM) that combines the global context modeling capability of transformers with the local feature extraction ability of CNNs for better feature fusion.

(2) The ATCM module utilizes an alternating transformer-CNN-transformer sequence, ensuring that global context guides local feature refinement, which in turn improves subsequent global modeling. To bridge transformers and CNNs, a transformer-to-CNN adapter and a CNN-to-transformer adapter are designed to enhance feature alignment. Advancing beyond simple stacking, the ATCM module deeply integrates transformers and CNNs, which enables it to fuse and refine multi-scale encoder features for providing stronger representations to the decoder.

(3) A knowledge distillation-guided low-light enhancement module (KDLEM) is designed to improve low-light adaptability, which is activated by an illumination gate. Specifically, the KDLEM module builds upon the illumination adaptive transformer (IAT) [28] to construct a teacher-student architecture. The teacher and the student use the original IAT and its simplified version to achieve comparable performance with reduced computational complexity. In this way, the overall network benefits from the rich prior knowledge of the teacher.

(4) To better validate the method, we construct a real-world low-light grasp detection dataset RLGD, containing over 70 objects captured under four distinct low-light intensities. Finally, experiments in datasets and real-world environments demonstrate the effectiveness of the proposed approach.

The remainder of this paper is structured as follows. Section 2 reviews the related work and Section 3 details the proposed method. The experimental results are presented in Section 4, followed by conclusions in Section 5.

2 Related Work

This section discusses the related work from two aspects: normal-light and low-light grasp detection.

2.1 Normal-Light Grasp Detection

Early studies predominantly relied on geometric feature matching. Although effective in structured environments, these methods faced limitations in generalizability due to their dependency on precise 3D models and sensitivity to non-rigid deformations.

The advent of deep learning revolutionized feature representation in grasp detection. Current deep learning-based approaches include candidate-based, direct regression, and pixel-wise grasp detection. The candidate-based methods first generate a large number of candidate grasps and then evaluate them using a predefined network to select the best grasp. Lenz et al. [6] pioneered the application of deep learning to robotic grasp detection. The method first generates numerous candidate grasps, then employs a two-stage cascaded network to sequentially evaluate and select the best grasp. Mahler et al. [7] developed a grasp quality CNN (GQ-CNN) which rapidly predicts grasp success probability from depth images. Guo et al. [8] introduced a novel hybrid deep architecture that integrates both visual and tactile perception for grasp detection. Meanwhile, Chu et al. [9] proposed a deep network capable of predicting multiple grasp candidates within the field of view. However, these methods are limited by their discrete sampling, which cannot guarantee full coverage of the grasp space and may thus miss the globally optimal grasp. Furthermore, such approaches exhibit computational complexity due to the sequential processing. To reduce computational time, the direct regression methods utilize the entire image to predict the best grasp. Redmon and Angelova [10] used a single-stage regression network to directly predict the best grasp without sliding windows or region proposals. In recent years, pixel-wise grasp detection has become a mainstream solution, as it generates grasp prediction of each pixel location. The dense prediction provides comprehensive coverage of all possible grasps on a target object, which offers strong interpretability while maintaining the efficiency advantage. Morrison et al. [11,29] first proposed a pixel-level real-time solution using the generative grasping CNN (GG-CNN). Following this idea, a series of studies emerge. Kumra et al. proposed GR-ConvNet [30],

a generative residual convolutional network that enables real-time grasp detection by leveraging residual connections for feature refinement. Yu et al. presented SE-ResUNet [31], which integrates residual blocks and channel attention within a UNet framework to enhance grasp detection precision. Cao et al. [32] developed a lightweight grasp detection network with a receptive field block and a multidimensional attention fusion network based on Gaussian kernel-based grasp representation. Liu et al. [12] addressed catastrophic forgetting in robotic grasping via a teacher-student architecture with selective knowledge distillation. knowledge distillation is also used by Nie et al. [33] and Peng et al. [13] to efficiently learn grasp detection features from large teacher model.

The aforementioned pixel-wise grasp detection methods predominantly utilize CNN-based frameworks. Considering that CNNs mainly model local instead of global information, transformers with superior capability in capturing long-range dependencies provide another choice. Wang et al. [15] proposed for the first time to integrate the Swin-Transformer architecture in [34] with grasp detection. Han et al. [35] introduced a vision-tactile transformer framework using TimeSformer [36] and ViViT [37] to predict safe grasping forces for deformable objects. Nevertheless, transformers still fall short in local feature extraction with higher computational complexity. Naturally, researchers focus on synergistic optimization of hybrid models based on CNN and transformer. A popular combination way is to embed transformer into the CNN encoder. Dong et al. [17] proposed an encoder-decoder model with a transformer network in the encoder and a fully convolutional network in the decoder respectively to improve model accuracy. Zuo et al. [16] designed a multiscale hybrid encoder that combines multiple CNN blocks with transformer layers to capture both high- and low-level features simultaneously. Yang et al. [38] developed a model to better capture global and local features by incorporating CNN-based modules and a hybrid backbone that combines RCrossFormer with Swin-Transformer. Another representative manner is parallel processing of CNN and transformer. Wan et al. [18] developed a hybrid CNN-transformer parallel architecture, which is embedded in both the encoder and decoder to achieve improved representation of local features and global information. However, the simple stacking of transformer and CNN is insufficient to provide robust representations for stable prediction.

2.2 Low-Light Grasp Detection

To solve the challenge of grasp detection in low-light conditions, multi-sensor fusion strategy is adopted. By integrating heterogeneous data sources, the loss of accuracy and robustness caused by visual degradation is compensated. Deng et al. [20] presented FuseGrasp, a fusion grasping framework combining millimeter-wave radar and RGB-D camera data. It exploits the radar signals to improve depth completion and material identification. Chen et al. [19] designed a teleoperation system using three infrared sensors mounted on the robot hand to provide proximity feedback, enabling online adjustments to pre-grasp poses by modeling potential energy based on sensor readings. Multiple sensors lead to engineering complexity and more researchers focus on depth camera-based solutions. Within this paradigm, an intuitive processing is to add an image enhancement pre-processing stage pre-trained on public datasets to deal with the low illumination. Ref. [21] presented a network with a pre-trained multi-scale context residual convolutional network (MCRNet) for image enhancement and a context residual convolutional network (CRNet) for detecting object planar grasping poses. Zhang et al. [39] also developed a pre-trained attention-based image enhancement network before the detection module to facilitate grasping tasks in dark environments. While these methods adopt pre-trained image enhancement to improve low-light image quality, the disconnection with grasp detection limits the system performance. Another means is to generate synthetic data under low-light conditions for network training. Luo et al. [25] synthesized data in Unity3D [40], which is used to train the proposed the deep visual servoing feature network (DVSFN). Further, researchers jointly

optimize enhancement and grasp detection to avoid the mutual disconnection, meanwhile, low-light training data is also generated. Niu et al. [26] designed a network that incorporates an unsupervised low-light enhancement module Zero-DCE [41] and residual blocks with coordinate attention, which is trained using data by consecutively adjusting the brightness with Gaussian noise in existing public datasets. Gao et al. [27] integrated the low-light appearance enhancement module (LAEM) with a CNN-based grasping pose estimation module. Also, a low-light dataset in virtual environments is built. By learning low-light features, the grasp detection performance is improved, however, the learned feature is still less adaptable. In this paper, the alternating transformer-CNN module and knowledge distillation-guided low-light enhancement are elaborately designed to boost the feature adaptability with good grasp detection results.

3 Methodology

In this section, we detail the proposed low-light aware hybrid grasp detection network, which is termed as LAH-Net. Pixel-wise grasp is represented as $g = \{x, y, q, \theta, w\}$ [11], where (x, y) represents the center coordinates of the grasp rectangle, q denotes the corresponding grasp quality, θ is the grasp orientation angle, and w is the width of the grasp rectangle. The output of the grasp detection network is denoted as $G = \{Q, S, C, W\}$, where Q, S, C, and W correspond to the feature maps of grasp quality, sine and cosine components of the grasp angle, and grasp width, respectively.

3.1 Overview

As illustrated in Fig. 1, the proposed LAH-Net model receives both RGB and depth images and predicts a pixel-wise grasp through an encoder-decoder architecture. Two core modules are provided: the alternating transformer-CNN module (ATCM) and the knowledge distillation-guided low-light enhancement module (KDLEM). The former locates between encoder and decoder to integrate alternating transformer and convolution blocks with cross-block adapters, enabling multi-scale feature fusion and long-range dependency modeling. And the latter connects encoder via the illumination gate under low-light conditions for image enhancement through a student network guided by a pre-trained network IAT [28].

KDLEM module is activated when the brightness L of the RGB image falls below a given threshold, enabling adaptation to both normal and low-light conditions. Let h_i denote the count at level i in grayscale image histogram h, $i = 0, 1, ..., 255$, and N_g the total number of pixels in the image. The brightness L of the RGB image is then computed as:

$$L = \frac{1}{255 \times N_g} \sum_{i=0}^{255} (i \times h_i) \tag{1}$$

The enhanced or normal RGB and the depth image are processed separately through two stem blocks. Each stem block consists of two 3×3 convolutions for feature alignment and dimension matching, and their outputs are fused through element-wise addition, which is fed into three CBR (Conv2d-BatchNorm2d-ReLU) residual blocks. Each CBR incorporates two groups of Conv2d, BatchNorm2d, and ReLU, progressively reducing resolution while enhancing feature representation. To strengthen feature propagation, the output of the second CBR residual block is combined with that of ATCM via a skip connection. The decoder consists of three CTBR (ConvTranspose2d-BatchNorm2d-ReLU) blocks, and it is responsible for recovering spatial resolution. Followed by four convolutional layers, the feature maps Q, S, C, and W are finally outputted.

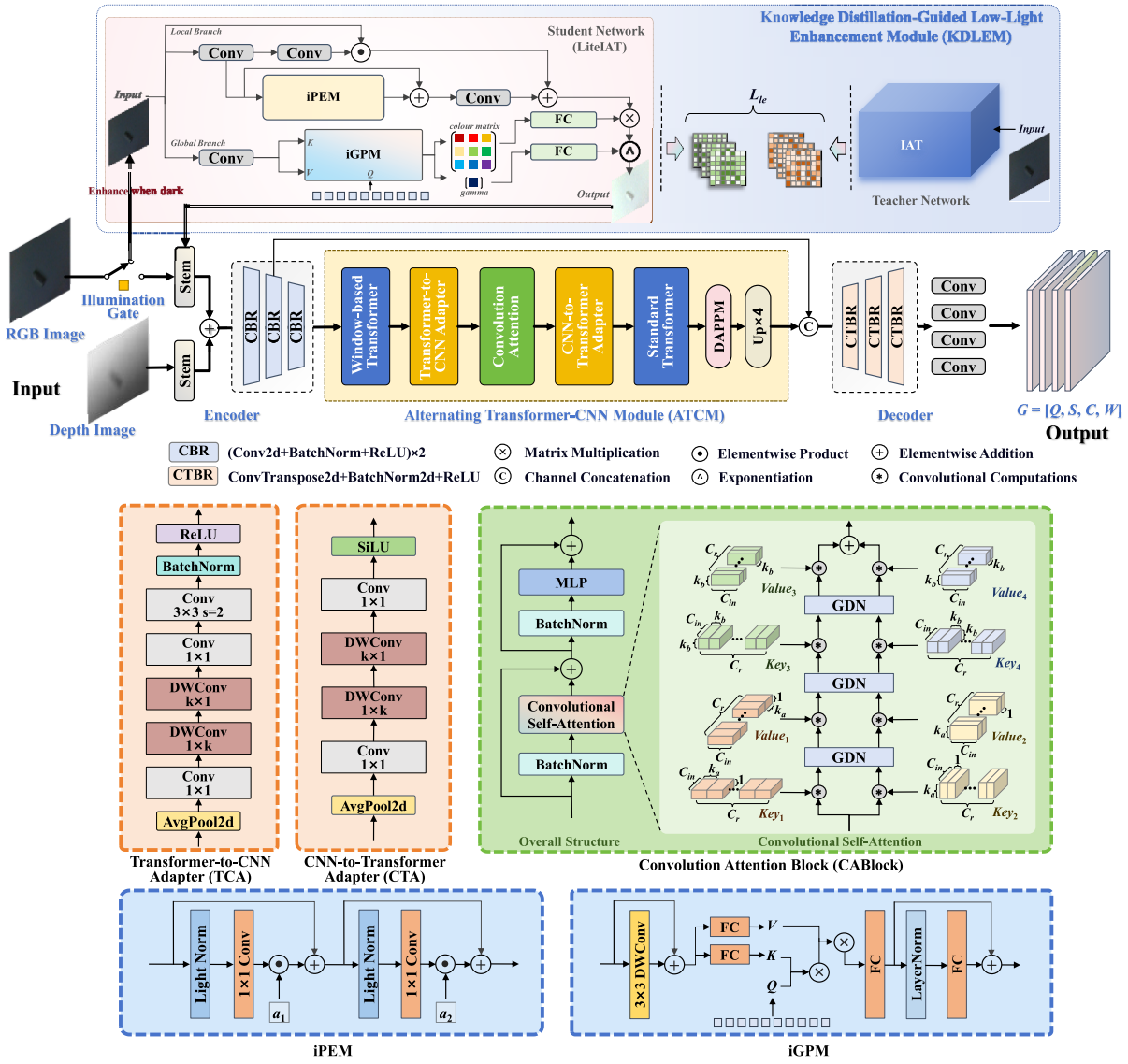

Figure 1: The framework of the proposed low-light aware hybrid network. Given a set of 224 × 224 RGB and depth images as inputs, the network predicts grasp feature maps ***G*** in an encoder-decoder manner with an alternating transformer-CNN module (ATCM) for better fusion of global and local features. Furthermore, an illumination gate is introduced for activating the knowledge distillation-guided low-light enhancement module (KDLEM) under low-light conditions. The ATCM module comprises a window-based transformer, a standard transformer, an improved convolutional attention block with a transformer-to-CNN adapter and a CNN-to-transformer adapter. The KDLEM module takes a pre-trained network IAT [28] as the basis to build a teacher-student architecture, where teacher and student networks adopt IAT and its simplified version with less computational complexity and competitive performance. The student network connects encoder through illumination gate for adaptive low-light enhancement.

3.2 Alternating Transformer-CNN Module

The alternating transformer-CNN module (ATCM) bridges the encoder and decoder with the purpose of achieving global context aggregation. Illumination degradation is spatially uneven, and simple feature fusion cannot cope with this complexity. Therefore, we propose an iterative global-local-global refinement mechanism. The first transformer extracts rough preliminary information about illumination by calculating

self-attention in non-overlapping windows. And it captures the overall distribution of illumination and provides context information for subsequent processing steps. Then, based on this overall information, the CNN block refines local geometric details under the guidance of this global prior. Its ability to capture local patterns helps recover fine structures that are important for grasp detection. Finally, the second transformer performs global grasping inference. After the local details are improved, it integrates them into a global context to produce consistent grasp predictions. This iterative global-local-global refinement enables multiple interactions between global and local information, which is especially important when illumination degradation varies across the image.

And since the attention mechanism is a key technology to reach this goal, it finds wide application in dense prediction tasks. For image segmentation, SCTNet [42] evolves the GPU-friendly attention (GFA) into the Conv-Former block by enlarging its learnable vectors to learnable kernels. To implement the attention mechanism, Conv-Former block uses efficient convolution operations and emulates the transformer structure to effectively learn the semantic information. Inspired by it, convolution attention block (CABlock) is designed for grasp detection task to enhance the attention mechanism by capturing crucial contextual information, as shown in Fig. 1. The key difference between our convolutional self-attention (CSA) in CABlock and the one in Conv-Former block lies in the convolutional composition of the branches. Specifically, we extend original strip convolution branches (orange and yellow) by adding a parallel set of standard convolutions (green and blue) to enrich the feature representation. Then, an additional grouped double normalization (GDN) layer [43] is inserted between the two convolution types to mitigate scale sensitivity between their inputs and outputs. To better illustrate the processing procedure of CSA, we define the basic function f_{CSA}. Let $x \in \mathbb{R}^{C_{in} \times H_c \times W_c}$ represent the input feature map, where C_{in}, H_c, and W_c are the channel dimensions, height and width of x, respectively. Let $K_{m,n} \in \mathbb{R}^{C_{in} \times C_r \times m \times n}$ and $K^T_{m,n} \in \mathbb{R}^{C_r \times C_{in} \times m \times n}$ be learnable key and value kernels, where C_r is the channel dimension of the intermediate feature map, and (m, n) is the learnable convolutional kernel size. Let $\varphi(\cdot)$ denote the function of GDN, which performs softmax across spatial dimensions and grouped L2 normalization along the channel dimension. Let $\circledast$ indicate convolutional computations. The f_{CSA} is then defined as:

$$f_{CSA}(x, m, n) = \varphi(x \circledast K_{m,n}) \circledast K^T_{m,n} \tag{2}$$

The processing pipeline of CSA is given as follows:

$$\mathrm{CSA}(X) = f_{CSA}(\varphi(f_{CSA}(X, k_a, 1)), k_b, k_b) + f_{CSA}(\varphi(f_{CSA}(X, 1, k_a)), k_b, k_b) \tag{3}$$

where X is the input of CSA, k_a and k_b denote the kernel sizes of strip convolutions and standard convolutions, respectively. The processing procedure of CABlock can be modeled as:

$$\begin{aligned} \hat{F}^{ca}_{in} &= \mathrm{CSA}(\mathrm{BN}(F^{ca}_{in})) + F^{ca}_{in} \\ F^{ca}_{out} &= \mathrm{MLP}\left(\mathrm{BN}\left(\hat{F}^{ca}_{in}\right)\right) + \hat{F}^{ca}_{in} \end{aligned} \tag{4}$$

where F^{ca}_{in} is the input of CABlock, BN is the BatchNorm layer, $\hat{F}^{ca}_{in}$ and F^{ca}_{out} are the output of CSA and MLP, respectively. MLP(·) is the function of multi-layer perceptron. The CABlock is designed to enhance local feature extraction while maintaining the computational efficiency. Its core idea is to introduce an attention-like mechanism through convolutional operations, allowing the network to focus on informative regions that are critical for grasp detection.

Despite its efficiency, the CABlock falls short in modeling long-range dependencies due to its limited receptive field. In contrast, transformer excels at global modeling, which leads us to introduce transformer-based blocks that effectively captures the global context necessary for accurate and robust grasp detection.

A window-based transformer block (WTBlock) [34] and a standard transformer block (STBlock) are placed before and after the CABlock, respectively, thereby constructing a coarse-to-fine feature refinement process. As the initial module of ATCM, the WTBlock establishes a global prior by computing self-attention within non-overlapping windows, balancing global modeling with computational efficiency. Following the WTBlock, the CABlock enhances local details under its global guidance. Then, the STBlock reintegrates these refined local features back into the global context. This pipeline forms a coherent, progressive refinement process, rather than a mere stack of CNN and transformer.

Transformer and CNN features differ significantly in distribution and granularity. Transformer features mainly capture global semantic information, while CNN features focus more on local textures. Direct concatenation or addition may lead to feature conflicts. To address this issue, we design two adapters to narrow the semantic gap between them. The transformer-to-CNN adapter (TCA) adapts transformer features to the CNN domain, while the CNN-to-transformer adapter (CTA) converts CNN features for compatibility. Each adapter begins with an average pooling layer for spatial information aggregation, followed by a standard convolution layer that performs local feature interaction and cross-channel fusion. Then, two cascaded depth-wise strip convolution layers are employed to approximate standard large-kernel convolution operations, achieving significant computational efficiency gains. Thereafter, the TCA adapter incorporates two standard convolution layers with BN and ReLU for stronger non-linearity, whereas the CTA adapter employs a single standard convolution with SiLU for efficiency in cross-dimensional attention. In the final stage of the ATCM, DAPPM [44] performs multi-scale contextual feature extraction and enhancement. The resultant features are subsequently fed into an upsampling module for spatial resolution restoration. From the analysis above, the ATCM achieves synergistic complementarity between global context modeling and local feature extraction through alternating feature extraction paradigms.

3.3 Knowledge Distillation-Guided Low-Light Enhancement Module

To enable robust grasping in low-light conditions, a knowledge distillation-guided low-light enhancement module (KDLEM) is incorporated within LAH-Net, as depicted in the upper portion of Fig. 1. In recent years, numerous effective general low-light enhancement methods have emerged. For instance, Illumination Adaptive Transformer (IAT) [28] utilizes a transformer-style structure to model parameters related to the image signal processing (ISP) pipeline, achieving competitive results. Inspired by its generality and flexibility, the teacher-student framework KDLEM is designed, where knowledge is distilled from the IAT teacher network to its simplified student network LiteIAT. The student network is formulated on the principles of the IAT model. Let F_C and F_{Raw} be the sRGB images and the raw-RGB images, respectively. Let M and A be the pixel-wise multiplication and addition parameters, $W_{c_i,c_j}(\cdot)$ be a 3×3 matrix modeling the white balance and colour space transform process (where c_i, c_j denote the three image channels), and γ represent the gamma correction. The enhancement process is then expressed as:

$$\begin{aligned} &F_{Raw} = F_C \odot M + A \\ &f = max\left(\sum_{c_j} W_{c_i,c_j}(F_{Raw})\right), 0)^{\gamma}, c_i, c_j \in \{r, g, b\} \end{aligned} \tag{5}$$

where $\odot$ denotes element-wise multiplication. To achieve the model in Eq. (5), the student network introduces several key simplifications over the teacher model. In the local branch, all pixel-wise enhancement modules (PEMs) preceding pixel-wise multiplication are removed, while those for pixel-wise addition are replaced with an integrated PEM (iPEM). In the global branch, the original global prediction module (GPM) is replaced with an integrated GPM (iGPM). As shown in the bottom portion of Fig. 1, the iPEM module processes the input through two sequential lightNorm [28] layers followed by convolutional

operations, where a_1 and a_2 represent two learnable small numbers introduced to ensure convergence stability. Compared to the original PEM, the iPEM module removes all depth-wise convolutions, as well as the GELU and 1×1 convolution before a_1 and a_2. The iGPM module is denoted as follows:

$$At\left(Q, K, V\right) = \text{SoftMax}\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{6}$$

where Q, K, and V are the query, key, and value matrices, respectively, and $\sqrt{d}$ is the scaling factor. This simplified version omits the final GELU and fully connected layers present in the original GPM. Collectively, these optimizations maintain core functionality while significantly reducing parameter count and computational complexity for real-time deployment.

3.4 Loss Function

During network training, the loss function comprises the grasp detection loss and the low-light enhancement loss, which are denoted as L_{gd} and L_{le}. To balance the loss weights across the two tasks, the method proposed in [45] is adopted, which utilizes homoscedastic task uncertainty to weigh losses. The overall loss is formulated as L:

$$L = \frac{1}{2\alpha_1^2}L_{gd} + \frac{1}{2\alpha_2^2}L_{le} + \log\alpha_1 + \log\alpha_2 \tag{7}$$

where α_1 and α_2 are the learnable observation noise parameters of the two tasks. For grasp detection, the MSE loss is employed to supervise the four output maps of the network against ground truth, which is presented as follows:

$$L_{gd} = \frac{1}{N}\sum_{i}^{N}\sum_{P\in\{Q,S,C,W\}}\left(P_i - \hat{P}_i\right)^2 \tag{8}$$

where N is the number of training samples. P and $\hat{P}$ represent the network output feature map and ground truth, respectively. To facilitate knowledge transfer from the pre-trained teacher network to the student network, L_{le} is designed as follows:

$$L_{le} = \frac{1}{N}\sum_{i}^{N}\left|F_{EM}^i - \hat{F}_{EM}^i\right| \tag{9}$$

where F_{EM}^i and $\hat{F}_{EM}^i$ are the outputs of the student and teacher networks, respectively, for the i-th sample.

4 Experiments

In this section, we validate the effectiveness of the proposed method on the Cornell grasping dataset [6], Jacquard V2 grasping dataset [46], a self-built low-light grasping dataset, and the real-world scenarios. The Cornell dataset comprises 885 RGB-D images with 640×480 resolution and 8019 grasp annotations. Following the prior works [10,11], we adopt the image-wise (IW) and object-wise (OW) evaluation metrics for comparative analysis. The Jacquard V2 dataset consists of 51k images and 11k object instances. It achieves improved accuracy by correcting annotation errors from the original version through a Human-In-The-Loop (HIL) process. 95% of the data is allocated as training set.

To evaluate algorithms in low-light conditions, we introduce a novel real-world low-light grasp detection dataset (RLGD) collected using an eye-in-hand Intel RealSense D435i RGB-D camera. The RLGD dataset contains over 70 common desktop objects, including stationery (such as pens, knives, and tape),

daily necessities (such as combs, cups, and bottles), fruit replicas (such as apples, oranges, and mangoes), tools (such as screwdrivers, tape measures, and peelers), and other everyday items (such as dolls, umbrellas, and Rubik's cubes). The objects are made of various materials, including plastic, metal, glass, rubber, and fabric. All the images are captured in a completely dark indoor environment, using a flashlight with a light intensity of four levels as the sole source of light. The original RGB and depth images are directly paired without any scaling or normalization processing. The dataset contains 41k sets of RGB-D images at 640 × 480 resolution, with 271k annotated grasp rectangles. To systematically evaluate illumination robustness, the dataset is partitioned into four sub-datasets in the order of decreased brightness (RLGD-A to RLGD-D). The numbers of the corresponding RGB-D images are approximately 9.4, 10.1, 11.0, and 11.3k, respectively, as illustrated in Fig. 2. The first, second, and third rows of Fig. 2 display the RGB images, their corresponding annotations, and the object instances within the images, respectively.

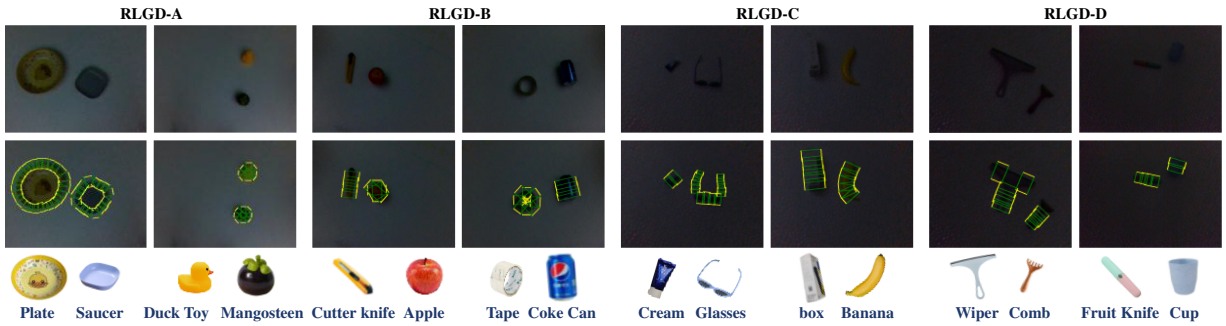

Figure 2: Four sub-datasets of the RLGD dataset with annotations.

Following [11,30], a successful grasp is defined by two criteria: (1) the angular difference between the predicted grasp and ground truth is less than 30°, and (2) the intersection over union (IOU) between the predicted grasp rectangle and ground truth must exceed 0.25. The selection of these metrics is mainly based on two considerations. Firstly, they can be directly compared with previous studies to ensure the fairness of the benchmark test. Secondly, these metrics are highly consistent with the actual deployment environment. The angle tolerance metric is used to assess the possible errors in the gripper orientation, while the IOU threshold ensures sufficient spatial overlap during the object lifting. Overall, these metrics can provide a reliable basis for evaluating the physical grasping success rate.

During training, all input images are resized to 224 × 224 pixels. All experiments are conducted on a system running Ubuntu 20.04, equipped with an NVIDIA RTX 4090D GPU, an Intel Core i9-14900K CPU, and 125 GB of memory. The model is implemented in Python 3.8 using the PyTorch 2.4.0 deep learning framework, with CUDA 12.4 and cuDNN 9.1. The Adam optimizer is used for training, with an initial learning rate of 1e−4. The model is trained for 50 epochs with a batch size of 8.

4.1 Comparison with Existing Methods

In this section, the proposed LAH-Net is compared with existing methods using datasets. For the Cornell dataset, the numbers from the original papers are directly used as the reference, which is consistent with the common practice in grasp detection literature [11,30]. This dataset has fixed train/test splits and widely accepted baseline results. For the Jacquard V2 and RLGD datasets, all methods are re-implemented based on a unified framework under the same experimental conditions. For methods with open-source code available, we adopt the official implementation and the input/output interfaces to match our grasp

representation. This design ensures that the performance comparison on Jacquard V2 and RLGD can reflect the algorithm's superiority rather than variations in training setup.

To demonstrate robust performance under normal-light conditions, the comparison of different methods on the Cornell dataset is conducted, as shown in Table 1. Except for [11,29] that use depth and [9] that uses RG-D, all other compared methods adopt RGB-D as input. The results show that LAH-Net achieves competitive performance on both IW and OW metrics, better than existing methods.

For further verification, the evaluation of the proposed LAH-Net is extended to the Jacquard V2 dataset, as shown in Table 2. Here, the semantic segmentation methods [42,47] are adapted by replacing their segmentation heads with LAH-Net's grasp detection decoder. LAH-Net demonstrates robust performance against existing methods, confirming its effectiveness under normal-light conditions.

Table 1: Comparison of different methods on the Cornell dataset.

Author	Method	Input	Accuracy (%)	
			IW	OW
Morrison et al. [11]	GG-CNN	D	73.0	69.0
Lenz et al. [6]	SAE-Net	RGB-D	73.9	75.6
Redmon and Angelova [10]	MultiGrasp	RGB-D	88.0	87.1
Yu et al. [48]	SKGNet	RGB-D	99.1	98.4
Kumra and kanan [49]	ResNet50-based	RGB-D	89.2	88.9
Asif et al. [50]	GraspNet	RGB-D	90.2	90.6
Chu et al. [9]	Multi grasp	RG-D	96.0	96.1
Morrison et al. [29]	GG-CNN2	D	84.0	82.0
Kumra et al. [30]	GR-ConvNet	RGB-D	97.7	96.6
Yu et al. [51]	TsGNet	RGB-D	93.1	93.0
Wang et al. [15]	TF-Grasp	RGB-D	98.0	96.7
Wu et al. [52]	PLG-Net	RGB-D	–	98.1
Ours	LAH-Net	RGB-D	99.4	98.8

Table 2: Comparison of different methods on the Jacquard V2 dataset.

Author	Method	Input	Accuracy (%)
Morrison et al. [11]	GG-CNN	RGB-D	87.56
Kumra et al. [30]	GR-ConvNet	RGB-D	94.22
Morrison et al. [29]	GG-CNN2	RGB-D	89.81
Yu et al. [31]	SE-ResUNet	RGB-D	93.60
Wan et al. [47]	SeaFormer	RGB	92.01
Wang et al. [15]	TF-Grasp	RGB-D	92.98
Yu et al. [48]	SKGNet	RGB-D	94.76
Xu et al. [42]	SCTNet	RGB-D	95.11
Ours	LAH-Net	RGB-D	96.05

Note: The results are reproduced using open-source codes.

To verify the effectiveness of the proposed method in low-light conditions, experiments on RLGD dataset are conducted. Each sub-dataset is split into a training set and a testing set in a randomized 9:1

ratio, with data augmentation techniques such as random rotation and zooming applied to the training data. Table 3 compares the performance of different methods on testing sets of RLGD-A, RLGD-B, RLGD-C, and RLGD-D, where each method is trained on the RLGD-A training set. Our method consistently ranks among the top across all illumination levels, showing stable performance as lighting degrades. With the dimming of illumination, the accuracy of existing methods suffers significant performance degradation. In contrast, our method is only slightly affected and keeps a higher accuracy. The results indicate the robustness of our method. Further, the results of different methods trained on the RLGD-D training set are summarized in Table 4. It is mentioning that the valuable features are more difficult to be extracted under darker conditions. Thus, some existing methods are heavily interfered (see the first column of Table 4), while our method maintains stable performance under these challenging conditions.

Table 3: Comparison of different methods (trained on RLGD-A) on testing sets of RLGD-A, RLGD-B, RLGD-C, and RLGD-D.

Author	Method	Accuracy (%)			
		RLGD-A	RLGD-B	RLGD-C	RLGD-D
Morrison et al. [11]	GG-CNN	64.20	59.66	57.13	48.89
Kumra et al. [30]	GR-ConvNet	90.64	80.78	67.67	65.61
Morrison et al. [29]	GG-CNN2	86.94	75.55	72.21	61.18
Yu et al. [31]	SE-ResUNet	94.89	87.22	77.48	70.82
Wang et al. [15]	TF-Grasp	92.06	85.61	66.76	57.47
Yu et al. [48]	SKGNet	95.43	92.05	79.30	73.03
Ours	LAH-Net	96.41	94.17	93.82	92.84

Note: The results are reproduced using open-source codes.

Table 4: Comparison of different methods (trained on RLGD-D) on testing sets of RLGD-A, RLGD-B, RLGD-C, and RLGD-D.

Author	Method	Accuracy (%)			
		RLGD-A	RLGD-B	RLGD-C	RLGD-D
Morrison et al. [11]	GG-CNN	10.77	17.00	65.21	80.19
Kumra et al. [30]	GR-ConvNet	17.52	35.31	83.38	95.49
Morrison et al. [29]	GG-CNN2	2.94	12.37	82.92	90.72
Yu et al. [31]	SE-ResUNet	43.74	70.22	93.73	96.55
Wang et al. [15]	TF-Grasp	7.83	20.82	87.65	94.08
Yu et al. [48]	SKGNet	32.75	64.39	91.19	96.20
Ours	LAH-Net	84.00	90.74	95.28	97.79

Note: The results are reproduced using open-source codes.

Fig. 3 visualizes t-SNE [53] embeddings of RLGD-A to RLGD-D sub-datasets before and after KDLEM processing. As shown in Fig. 3a, embeddings of original RGB images exhibit significant distribution shifts across sub-datasets due to illumination variations, which explains the limited generalization of models trained on individual sub-datasets. In contrast, Fig. 3b demonstrates that KDLEM-processed features form overlapping distributions in the embedding space, confirming the efficacy of the module in mitigating illumination discrepancies and enhancing adaptability to darker environments.

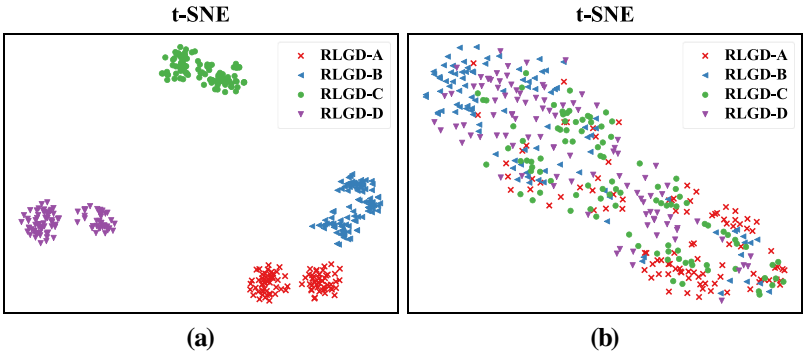

Figure 3: The t-SNE embeddings of sub-datasets before and after KDLEM processing. (**a**) Before KDLEM. (**b**) After KDLEM.

Fig. 4 illustrates the result images of LAH-Net trained and tested on the RLGD dataset with a 9:1 split. The first to fourth columns correspond to test samples from different sub-datasets of RLGD, with four rows showing the RGB images, depth images, grasp quality heatmaps, and grasp angle maps, respectively. As shown, the predicted grasp rectangles are visualized on the depth maps. The results show that LAH-Net can accurately generate physically feasible grasp rectangles, maintaining robust performance even under challenging illumination conditions. For a more intuitive comparison of the methods, Fig. 5 presents the results across two representative scenarios. Each group contains four algorithm outputs: GR-ConvNet, TF-Grasp, SKGNet, and LAH-Net. The first, second, and third rows show the RGB images, depth images with grasp rectangles, and grasp quality heatmaps, respectively. Every method outputs up to five grasp rectangles in distinct colors, representing the top five quality scores. Only predictions exceeding a quality threshold are retained, and the highest-scoring one is selected for physical execution. LAH-Net achieves more accurate results compared to other methods.

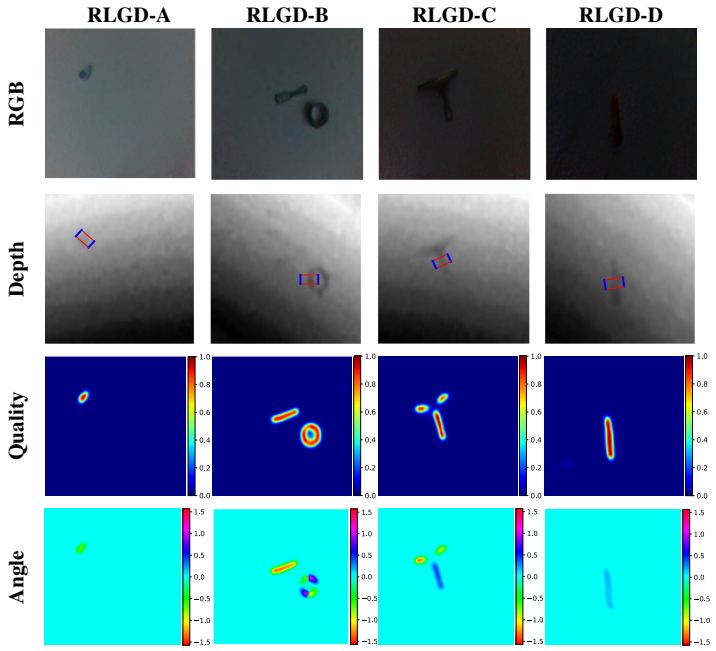

Figure 4: The grasp detection results of LAH-Net tested on RLGD dataset.

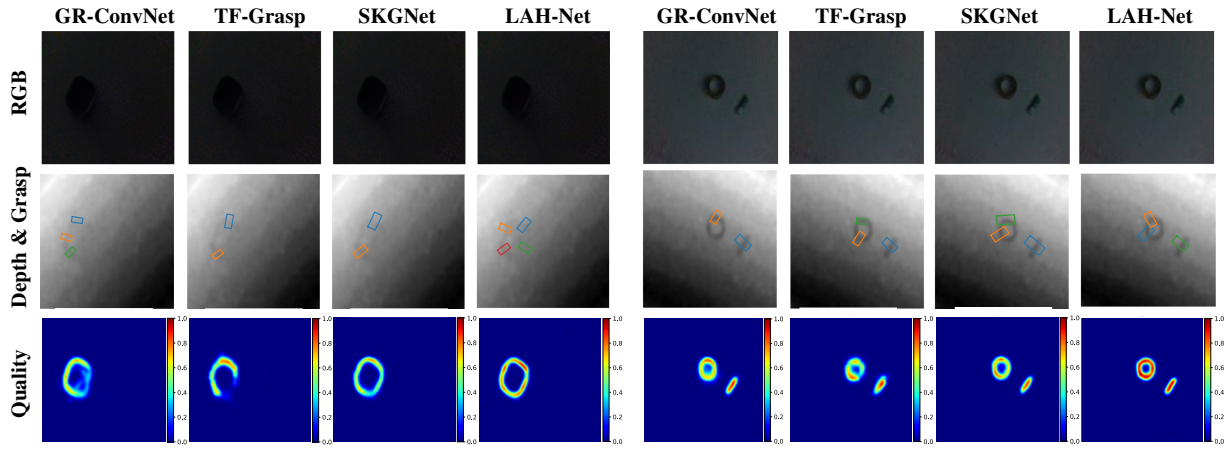

Figure 5: The grasp detection results of different methods on RLGD.

To illustrate the adaptability to diverse objects of LAH-Net, Fig. 6 presents its results on various objects from the RLGD test set. The results demonstrate the capability of the proposed method to handle various objects.

Figure 6: The grasp detection results on RLGD test set.

4.2 Ablation Studies

(1) *Ablation of LAH-Net*. To validate the performance of our proposed LAH-Net, its two variants are taken into account according to whether ATCM and KDLEM modules are adopted. LAH-Net-I incorporates KDLEM but removes ATCM, while LAH-Net-II retains ATCM but excludes KDLEM. As summarized in Table 5 on the RLGD dataset, the absence of either module leads to a significant degradation in grasp detection accuracy. These results confirm the complementary roles of both components, with the ATCM module enabling robust feature representation while the KDLEM module ensures reliable performance in low-light environments. To validate the efficacy of the ATCM module, five network variants (LAH-Net-III to LAH-Net-VII) are designed and trained on the Jacquard V2 dataset, as presented in the lower section of Table 5. LAH-Net-III is derived from LAH-Net by removing the TCA and CTA adapters. Building upon this, LAH-Net-IV replaces the WTBlock and STBlock in LAH-Net-III with standard convolutions. Besides, LAH-Net-V and LAH-Net-VI modify the CABlock, replacing it with the Conv-Former block and a standard convolution, respectively. Lastly, LAH-Net-VII substitutes

the DAPPM module with a standard convolution. From the results of LAH-Net-III and LAH-Net, one can see that removing the adapters leads to a drop in grasp detection accuracy, validating their essential role in bridging the convolutional and transformer modules. Compared to LAH-Net-III, LAH-Net-IV performs worse, which proves that transformer-based modules enhance the performance by capturing long-range dependencies. It is seen that LAH-Net is better than LAH-Net-V and LAH-Net-VI, which shows that the CABlock excels at capturing crucial contextual information. Finally, the results from LAH-Net-VII and LAH-Net substantiate the effectiveness of the DAPPM module in multi-scale contextual feature extraction.

Table 5: Results of ablation experiments of LAH-Net on the RLGD and the Jacquard V2 dataset.

Dataset	Method	ATCM						KDLEM	Accuracy (%)
		TCA	CTA	WTBlock	STBlock	CABlock	DAPPM		
RLGD	LAH-Net-I	–	–	–	–	–	–	√	95.37
	LAH-Net-II	√	√	√	√	√	√	–	95.90
	LAH-Net	√	√	√	√	√	√	√	96.50
Jacquard V2	LAH-Net-III	–	–	√	√	√	√	–	95.78
	LAH-Net-IV	–	–	–	–	√	√	–	95.35
	LAH-Net-V	√	√	√	√	–	√	–	95.51
	LAH-Net-VI	√	√	√	√	–	√	–	95.20
	LAH-Net-VII	√	√	√	√	√	–	–	95.54
	LAH-Net	√	√	√	√	√	√	–	96.05

(2) *Ablation of KDLEM*. To validate the effectiveness of the KDLEM module, we designed two variants, KDLEM-I and KDLEM-II. KDLEM-I directly employs a frozen-weight IAT as the pre-processing module for the entire network. In contrast, KDLEM-II incorporates IAT as the enhancement module that participates in joint network training and weight updates. As shown in Table 6, directly using the IAT module as a pre-processor is suboptimal, since its generic enhancements are not optimized for the specialized feature space of grasp detection. In addition, the evaluation of KDLEM-II and KDLEM indicates that IAT and LiteIAT yield similar accuracy. A key advantage of LiteIAT is its efficiency, requiring only 11.8% of the FLOPs and 24.8% of the parameters relative to IAT. The above results collectively demonstrate the effectiveness of KDLEM in balancing performance and efficiency.

Table 6: Results of ablation experiments of the KDLEM module on the RLGD dataset.

Method	IAT (Frozen)	IAT	LiteIAT	Module FLOPs (G)	Module Params (K)	Accuracy (%)
KDLEM-I	√	–	–	1.10	91.15	95.44
KDLEM-II	–	√	–	1.10	91.15	96.50
KDLEM	–	–	√	0.13	22.58	96.50

(3) *Ablation of different architectural arrangements for the ATCM module*. To verify the effectiveness of the alternating transformer-CNN-transformer design in the ATCM module, we conduct a structural ablation experiment on the RLGD-D subset. The definitions and results of each variant are shown in Table 7. The ATCM-I variant only retains WTBlock and STBlock, removing CABlock. It achieves an accuracy of 96.29%, which is 1.50% lower than the baseline, demonstrating the necessity of local detail refinement provided by the CABlock. ATCM-II repeats the CABlock twice and removes all

transformer modules. Its accuracy is 1.86% lower than the baseline, indicating the importance of the global illumination prior. ATCM-III processes features in parallel using WTBlock and CABlock, and then merges the results before feeding them to STBlock. This variant achieves the second-best performance, suggesting that combing both global and local features is beneficial. However, a single parallel fusion is still not as effective as iterative interaction. ATCM-IV follows the sequence of CABlock, WTBlock, and STBlock. Its accuracy is higher than that of pure CNN variant (ATCM-II) but lower than that of pure transformer variant (ATCM-I). This indicates that, even with subsequent global integration, starting with local feature extraction cannot fully compensate for the lack of initial global prior information. The ATCM-V variant follows the sequence of WTBlock and CABlock, while the ATCM-VI variant follows the sequence of CABlock and WTBlock. Neither variant includes the final STBlock, and both are significantly lower than the baseline. This further indicates that regardless of the sequence, single-stage fusion is difficult to achieve ideal results in extremely low light conditions. In conclusion, the complete transformer-CNN-transformer design achieves the highest accuracy among all variants, verifying the effectiveness and necessity of the iterative global-local-global mechanism.

Table 7: Comparison of different architectural arrangements for the ATCM module on the RLGD-D dataset.

Variant	Description	Accuracy (%)
ATCM	WTBlock (T) → CABlock (C) → STBlock (T)	97.79
ATCM-I	Transformer only: WTBlock (T) → STBlock (T)	96.29
ATCM-II	CNN only: CABlock (C) × 2	95.93
ATCM-III	WTBlock (T)//CABlock (C) → STBlock (T)	96.73
ATCM-IV	CABlock (C) → WTBlock (T) → STBlock (T)	96.02
ATCM-V	WTBlock (T) → CABlock (C)	95.84
ATCM-VI	CABlock (C) → WTBlock (T)	95.76

4.3 Robotic Grasping in Real-World Scenarios

In this section, robotic grasping experiments in real-world nighttime scenarios are conducted to validate the effectiveness of the proposed LAH-Net. For the maps output by the network, the best grasp $g^{\star}$ is determined by the peak in the grasp quality map, with its angle and width taken directly from the associated maps at that location. This image-space grasp is then projected to the 6-D pose g_b in the robot coordinate system using the extrinsic parameters of the camera. Then, g_b is executed by our robotic grasping system, which comprises an Elephant Robotics MyCobot Pro630 manipulator equipped with a two-finger parallel gripper and an eye-in-hand Intel RealSense D435i camera. For fair evaluation, 20 distinct objects are selected with 10 grasp attempts per object. The robot picks up the object and places it at the target location. As shown in Table 8, comparative evaluations are performed against GG-CNN, GR-ConvNet, TF-Grasp, and SKGNet, with all models trained on the RLGD dataset using identical configurations. The results demonstrate that LAH-Net achieves the highest success rate, confirming the robustness and stability of our method. To illustrate the practical performance in real-world scenarios, Fig. 7 visualizes the robotic grasping processes under low-light conditions. Fig. 7a,b present the RGB images, depth images with grasp rectangles, and grasp quality maps for a banana and a box, respectively, while Fig. 7c,d depict the sequential execution of the grasp. Notably, the success of these grasps under challenging illumination attests to the effectiveness of our method in real-world low-light scenarios.

Table 8: Grasping success rate in real-world scenarios.

Author	Method	Success Rate (%)
Morrison et al. [11]	GG-CNN	84.5
Kumra et al. [30]	GR-ConvNet	89.0
Wang et al. [15]	TF-Grasp	90.0
Yu et al. [48]	SKGNet	93.0
Our method	LAH-Net	95.5

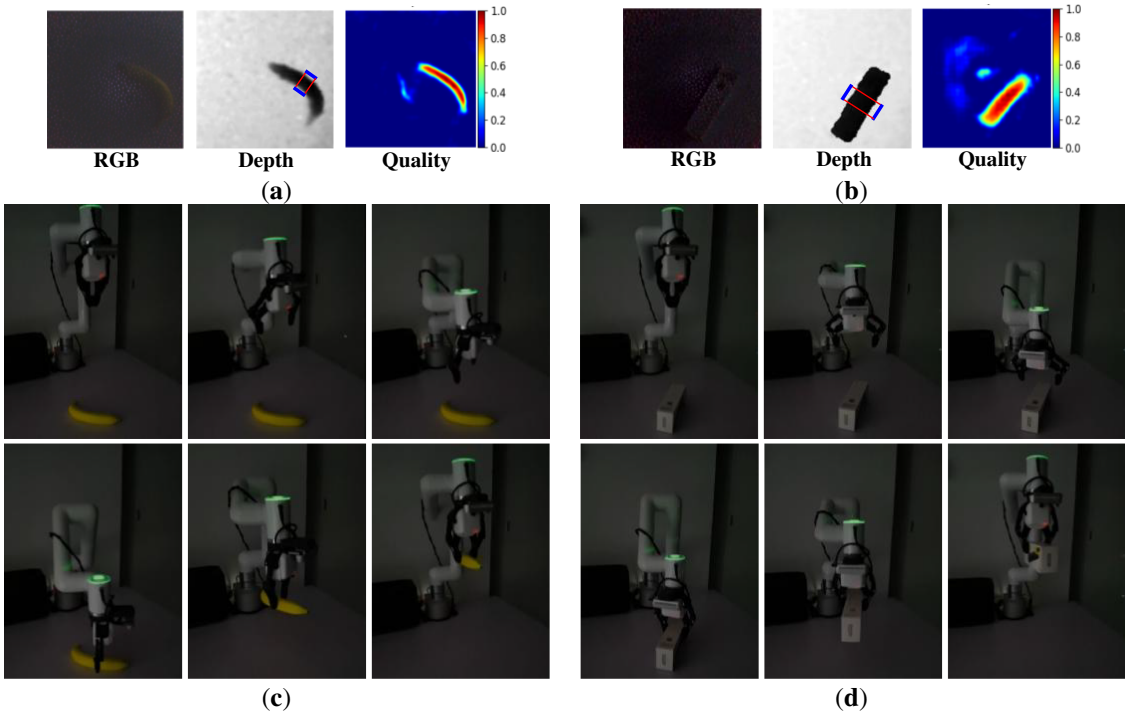

Figure 7: Visualization of robotic grasping process in low-light scenarios. (**a**) Grasp detection results of a banana. (**b**) Grasp detection results of a box. (**c**) Grasping process of a banana. (**d**) Grasping process of a box.

4.4 Discussion

The experimental results provide several insights into why our method performs better under low-light conditions.

1. Table 3 shows that when models are trained on the brightest training set and tested on darker testing sets, existing methods suffer large accuracy drops. In contrast, LAH-Net maintains high accuracy even on the darkest testing set. We attribute this to two factors. First, KDLEM enhances image quality and recovers structural details. Second, ATCM iteratively refines both global and local features. Together, they make the model less sensitive to the loss of low-level visual information.
2. The ablation study shows that our transformer-CNN-transformer design outperforms other variants. This supports our design rationale. The first transformer captures coarse illumination priors. The CNN then refines local details under those priors. The final transformer integrates everything for global grasp reasoning. This iterative refinement is particularly useful when lighting is uneven.

3. The t-SNE visualization in Fig. 3 shows that KDLEM aligns feature distributions across different light levels. This reduces domain shift and improves generalization. The proposed method is therefore especially helpful when training and test conditions differ.

5 Conclusion

This article introduces a novel low-light aware hybrid network LAH-Net for robotic grasping. Specifically, the alternating transformer-CNN module (ATCM) synergizes the long-range dependency modeling capacity of transformers with the local feature extraction proficiency of convolutions, enhanced by a cross-block adapter for feature transition. Then, the knowledge distillation-guided low-light enhancement module (KDLEM) enhances adaptability to low-light conditions while minimizing computational cost through a teacher-student framework. To facilitate robust algorithm validation, we also present the RLGD dataset, containing over 70 objects spanning four low-light illumination. Experiments demonstrate that LAH-Net achieves superior performance compared to existing methods across the RLGD, Cornell, and Jacquard V2 datasets. Finally, real-world tests confirm our method's effectiveness in practical low-light scenarios, which provides a robust solution for vision-guided manipulation under degraded conditions.

Although the experimental results demonstrate the effectiveness of the proposed method, there are still several limitation. First, the RLGD dataset only contains a limited number of object categories and lacks background diversity with a single sensor. In addition, the illumination variations are limited to brightness and do not cover other factors. As a result, the generalization of the model to more diverse real-world scenarios still needs further validation.

Future research could explore several directions. One is developing more robust architectures that handle multiple visual degradations simultaneously. Another is building more comprehensive benchmarks that cover a wider range of lighting conditions and scene complexities.

Acknowledgement: Not applicable.

Funding Statement: This work was supported in part by the Financial Program of BJAST under Grants 26CE-BGS-19, 26CB012-03, and in part by the National Natural Science Foundation of China under Grants 62536001.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Yingying Yu and Jun Yuan; methodology, Yingying Yu and Jun Yuan; software, Yingying Yu and Jun Yuan; validation, Yingying Yu and Jun Yuan; formal analysis, Tong Liu; investigation, Tong Liu; resources, Yingying Yu; data curation, Yingying Yu; writing—original draft preparation, Yingying Yu; writing—review and editing, Yingying Yu and Jun Yuan; visualization, Yingying Yu and Jun Yuan; supervision, Yingying Yu and Jun Yuan; project administration, Yingying Yu; funding acquisition, Yingying Yu. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data presented in this study are available on request from the corresponding author. The RLGD dataset is available at https://github.com/EvelynDev25/RLGD_Dataset.

Ethics Approval: Not applicable.

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

References

1. Lu R, Xu M, Zhu W, Yang Z, Chao D, Zhang Y, et al. A robot grasp detection method based on neural architecture search and its interpretability analysis. Comput Mater Contin. 2026;87(1):1–10. doi:10.32604/cmc.2025.073442.
2. Zhang Q, Hu B, Qin J, Duan J, Zhou Y. A low-collision and efficient grasping method for manipulator based on safe reinforcement learning. Comput Mater Contin. 2025;83(1):1257–73. doi:10.32604/cmc.2025.059955.

3. Bakirci M, Demiray A. Tracking robotic arms with YOLO11 for smart automation in industry 4.0. In: Proceedings of the 2025 International Russian Smart Industry Conference (SmartIndustryCon); 2025 Mar 24–28; Sochi, Russian. p. 64–70. doi:10.1109/SmartIndustryCon65166.2025.10985968.
4. An S, Meng Z, Tang C, Zhou Y, Liu T, Ding F, et al. Dexterous manipulation through imitation learning: a survey. IEEE Trans Autom Sci Eng. 2026;23(3):1760–92. doi:10.1109/TASE.2025.3646183.
5. Newbury R, Gu M, Chumbley L, Mousavian A, Eppner C, Leitner J, et al. Deep learning approaches to grasp synthesis: a review. IEEE Trans Robot. 2023;39(5):3994–4015. doi:10.1109/tro.2023.3280597.
6. Lenz I, Lee H, Saxena A. Deep learning for detecting robotic grasps. Int J Robot Res. 2015;34(4–5):705–24. doi:10.1177/0278364914549607.
7. Mahler J, Liang J, Niyaz S, Laskey M, Doan R, Liu X, et al. Dex-net 2.0: deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In: Proceedings of the Robotics: Science and Systems XIII; 2017 Jul 12–16; Cambridge, MA, USA. doi:10.15607/rss.2017.xiii.058.
8. Guo D, Sun F, Liu H, Kong T, Fang B, Xi N. A hybrid deep architecture for robotic grasp detection. In: Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA); 2017 May 29–Jun 3; Singapore. p. 1609–14. doi:10.1109/ICRA.2017.7989191.
9. Chu FJ, Xu R, Vela PA. Real-world multiobject, multigrasp detection. IEEE Robot Autom Lett. 2018;3(4):3355–62. doi:10.1109/LRA.2018.2852777.
10. Redmon J, Angelova A. Real-time grasp detection using convolutional neural networks. In: Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA); 2015 May 26–30; Seattle, WA, USA. p. 1316–22. doi:10.1109/ICRA.2015.7139361.
11. Morrison D, Leitner J, Corke P. Closing the loop for robotic grasping: a real-time, generative grasp synthesis approach. In: Proceedings of the Robotics: Science and Systems XIV; 2018 Jun 26–30; Pittsburgh, PA, USA. doi:10.15607/rss.2018.xiv.021.
12. Liu J, Xie J, Huang S, Wang C, Zhou F. Continual learning for robotic grasping detection with knowledge transferring. IEEE Trans Ind Electron. 2024;71(9):11019–27. doi:10.1109/TIE.2023.3342312.
13. Peng L, Cai R, Xiang J, Zhu J, Liu W, Gao W, et al. LiteGrasp: a light robotic grasp detection via semi-supervised knowledge distillation. IEEE Robot Autom Lett. 2024;9(9):7995–8002. doi:10.1109/LRA.2024.3436336.
14. Sarker PK, Zhao Q. Enhanced visible-infrared person re-identification based on cross-attention multiscale residual vision transformer. Pattern Recognit. 2024;149(1):110288. doi:10.1016/j.patcog.2024.110288.
15. Wang S, Zhou Z, Kan Z. When transformer meets robotic grasping: exploits context for efficient grasp detection. IEEE Robot Autom Lett. 2022;7(3):8170–7. doi:10.1109/LRA.2022.3187261.
16. Zuo G, Shen Z, Yu S, Luo Y, Zhao M. HBGNet: robotic grasp detection using a hybrid network. IEEE Trans Instrum Meas. 2025;74:2503109. doi:10.1109/TIM.2024.3522557.
17. Dong M, Bai Y, Wei S, Yu X. Robotic grasp detection based on transformer. In: Proceedings of the 15th International Conference on Intelligent Robotics and Applications; 2022 Aug 1–3; Harbin, China. p. 437–48. doi:10.1007/978-3-031-13841-6_40.
18. Wan Q, Ning S, Tan H, Wang Y, Duan X, Li Z, et al. FFBGNet: full-flow bidirectional feature fusion grasp detection network based on hybrid architecture. IEEE Robot Autom Lett. 2025;10(2):971–8. doi:10.1109/LRA.2024.3511410.
19. Chen N, Tee KP, Chew CM. Teleoperation grasp assistance using infra-red sensor array. Robotica. 2015;33(4):986–1002. doi:10.1017/s0263574714000733.
20. Deng H, Xue T, Chen H. FuseGrasp: radar-camera fusion for robotic grasping of transparent objects. IEEE Trans Mob Comput. 2025;24(8):7028–41. doi:10.1109/TMC.2025.3547371.
21. Xu F, Zhu Z, Feng C, Leng J, Zhang P, Yu X, et al. An object planar grasping pose detection algorithm in low-light scenes. Multimed Tools Appl. 2025;84(9):5583–604. doi:10.1007/s11042-024-19128-5.
22. Jiang Y, Li L, Zhu J, Xue Y, Ma H. DEANet: decomposition enhancement and adjustment network for low-light image enhancement. Tsinghua Sci Technol. 2023;28(4):743–53. doi:10.26599/TST.2022.9010047.
23. Jiang X, Gao N, Dou H, Zhang X, Zhong X, Deng Y, et al. Global modeling matters: a fast, lightweight and effective baseline for efficient image restoration. IEEE Trans Image Process. 2026;35:2740–54. doi:10.1109/TIP.2026.3671691.

24. Shang X, An N, Zhang S, Ding N. Toward robust and efficient low-light image enhancement: progressive attentive retinex architecture search. Tsinghua Sci Technol. 2023;28(3):580–94. doi:10.26599/TST.2022.9010017.
25. Luo J, Zhang Z, Wang Y, Feng R. Robot closed-loop grasping based on deep visual servoing feature network. Actuators. 2025;14(1):25. doi:10.3390/act14010025.
26. Niu M, Lu Z, Chen L, Yang J, Yang C. VERGNet: visual enhancement guided robotic grasp detection under low-light condition. IEEE Robot Autom Lett. 2023;8(12):8541–8. doi:10.1109/LRA.2023.3330664.
27. Gao Y, Chen L, Liu J. VAEPose: 6D pose estimation with visual appearance enhancement for low-light conditions. In: Proceedings of the 2024 IEEE International Conference on Industrial Technology (ICIT); 2024 Mar 25–27; Bristol, UK. p. 1–7. doi:10.1109/ICIT58233.2024.10540716.
28. Cui Z, Li K, Gu L, Su S, Gao P, Jiang Z, et al. You only need 90K parameters to adapt light: a light weight transformer for image enhancement and exposure correction. In: Proceedings of the British Machine Vision Conference (BMVC); 2022 Nov 21–24; London, UK.
29. Morrison D, Corke P, Leitner J. Learning robust, real-time, reactive robotic grasping. Int J Robot Res. 2020;39 (2–3):183–201. doi:10.1177/0278364919859066.
30. Kumra S, Joshi S, Sahin F. Antipodal robotic grasping using generative residual convolutional neural network. In: Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2020 Oct 25–29; Las Vegas, NV, USA. p. 9626–33. doi:10.1109/IROS45743.2020.9340777.
31. Yu S, Zhai DH, Xia Y, Wu H, Liao J. SE-ResUNet: a novel robotic grasp detection method. IEEE Robot Autom Lett. 2022;7(2):5238–45. doi:10.1109/LRA.2022.3145064.
32. Cao H, Chen G, Li Z, Feng Q, Lin J, Knoll A. Efficient grasp detection network with Gaussian-based grasp representation for robotic manipulation. IEEE/ASME Trans Mechatron. 2023;28(3):1384–94. doi:10.1109/TMECH.2022.3224314.
33. Nie H, Zhao Z, Chen L, Lu Z, Li Z, Yang J. Smaller and faster robotic grasp detection model via knowledge distillation and unequal feature encoding. IEEE Robot Autom Lett. 2024;9(8):7206–13. doi:10.1109/LRA.2024.3421790.
34. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, et al. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct 10–17; Montreal, QC, Canada. p. 9992–10002. doi:10.1109/iccv48922.2021.00986.
35. Han Y, Yu K, Batra R, Boyd N, Mehta C, Zhao T, et al. Learning generalizable vision-tactile robotic grasping strategy for deformable objects via transformer. IEEE/ASME Trans Mechatron. 2025;30(1):554–66. doi:10.1109/TMECH.2024.3400789.
36. Bertasius G, Wang H, Torresani L. Is space-time attention all you need for video understanding? ICML. 2021;2(3):4.
37. Arnab A, Dehghani M, Heigold G, Sun C, Lucic M, Schmid C. ViViT: a video vision transformer. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct 10–17; Montreal, QC, Canada. p. 6816–26. doi:10.1109/iccv48922.2021.00676.
38. Yang L, Zhang C, Liu G, Zhong Z, Li Y. A model for robot grasping: integrating transformer and CNN with RGB-D fusion. IEEE Trans Consum Electron. 2024;70(2):4673–84. doi:10.1109/TCE.2024.3403848.
39. Zhang K, Li Q, Liu K, Zhang M, Zhu Z, Feng C. AM-GPD: manipulator grasping pose detector based on attention mechanism in dark light scene. In: Proceedings of the 2023 China Automation Congress (CAC); 2023 Nov 17–19; Chongqing, China. p. 2281–6. doi:10.1109/CAC59555.2023.10451789.
40. Horváth D, Erdős G, Istenes Z, Horváth T, Földi S. Object detection using Sim2Real domain randomization for robotic applications. IEEE Trans Robot. 2023;39(2):1225–43. doi:10.1109/TRO.2022.3207619.
41. 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 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun 13–19; Seattle, WA, USA. p. 1777–86. doi:10.1109/cvpr42600.2020.00185.
42. Xu Z, Wu D, Yu C, Chu X, Sang N, Gao C. SCTNet: single-branch CNN with transformer semantic information for real-time segmentation. Proc AAAI Conf Artif Intell. 2024;38(6):6378–86. doi:10.1609/aaai.v38i6.28457.
43. Wang J, Gou C, Wu Q, Feng H, Han J, Ding E, et al. RTFormer: efficient design for real-time semantic segmentation with transformer. Adv Neural Inf Process Syst. 2022;35:7423–36. doi:10.52202/068431-0539.

44. Pan H, Hong Y, Sun W, Jia Y. Deep dual-resolution networks for real-time and accurate semantic segmentation of traffic scenes. IEEE Trans Intell Transp Syst. 2023;24(3):3448–60. doi:10.1109/TITS.2022.3228042.
45. Cipolla R, Gal Y, Kendall A. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 7482–91. doi:10.1109/CVPR.2018.00781.
46. Li Q, Yuan S. Jacquard V2: refining datasets using the human in the loop data correction method. In: Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA); 2024 May 13–17; Yokohama, Japan. p. 7932–8. doi:10.1109/ICRA57147.2024.10611652.
47. Wan Q, Huang Z, Lu J, Yu G, Zhang L. Seaformer: squeeze-enhanced axial transformer for mobile semantic segmentation. In: Proceedings of the 11th International Conference on Learning Representations; 2023 May 1–5; Kigali, Rwanda. p. 1–19.
48. Yu S, Zhai DH, Xia Y. SKGNet: robotic grasp detection with selective kernel convolution. IEEE Trans Autom Sci Eng. 2023;20(4):2241–52. doi:10.1109/TASE.2022.3214196.
49. Kumra S, Kanan C. Robotic grasp detection using deep convolutional neural networks. In: Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2017 Sep 24–28; Vancouver, BC, Canada. p. 769–76. doi:10.1109/IROS.2017.8202237.
50. Asif U, Tang J, Harrer S. GraspNet: an efficient convolutional neural network for real-time grasp detection for low-powered devices. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence; 2018 Jul 13–19; Stockholm, Sweden. p. 4875–82. doi:10.24963/ijcai.2018/677.
51. Yu Y, Cao Z, Liu Z, Geng W, Yu J, Zhang W. A two-stream CNN with simultaneous detection and segmentation for robotic grasping. IEEE Trans Syst Man Cybern Syst. 2022;52(2):1167–81. doi:10.1109/TSMC.2020.3018757.
52. Wu Y, Fu Y, Wang S. Information-theoretic exploration for adaptive robotic grasping in clutter based on real-time pixel-level grasp detection. IEEE Trans Ind Electron. 2024;71(3):2683–93. doi:10.1109/TIE.2023.3270537.
53. Maaten L, Hinton G. Visualizing data using t-SNE. J Mach Learn Res. 2008;9(11):2579–605.