Available online at **www.joac.info**

ISSN: 2278-1862

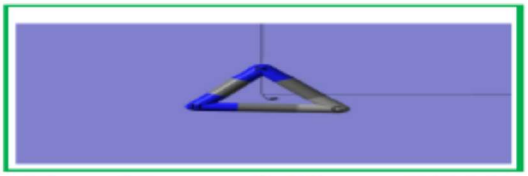
Journal of Applicable Chemistry

2025, 14 (2): 395-470

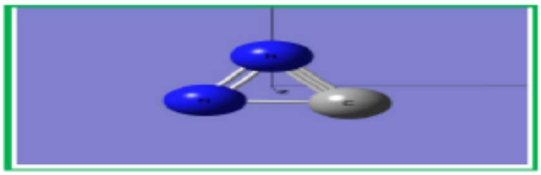
(International Peer Reviewed Journal)

New Chemistry News

$^{-}N=C=N^{-}$



New News of Chem (NNC)



ChemNewsNew (CNN)

CNN – 66C

Iam (Intelligence Augmented /Assisted Method(s))

Transformers --architectures & Fits (2025)

Information Source	sciencedirect.com;	
S. Narasinga Rao M D *Associate Professor,* *Emergency Medicine dept.,* *Andhra Medical College,* *King George Hospital* *Visakhapatnam, A.P., India*	*K. Somasekhara Rao, Ph D* *Dept. of Chemistry,* *Acharya Nagarjuna Univ.,* *Dr. M.R.Appa Rao Campus,* *Nuzvid-521 201, India*	*R. Sambasiva Rao, Ph D* *Dept. of Chemistry,* *Andhra University,* *Visakhapatnam 530 003,* *India*
snrnaveen007@gmail.com (+91 98 48 13 67 04)	sr_kaza1947@yahoo.com (+91 98 48 94 26 18)	rsr.chem@gmail.com (+91 99 85 86 01 82)

Conspectus: In the year 2017, Ashish Vaswani et al. published a paper entitled "Attention is All You Need". It revolutionised sequence data modelling. The new model gained popularity as a Transformer net (TransF Net) or Transformer neural network (TransF NN). The two important modules are attention layer and MLP-NN to carry out Natural Language processing (NLP). The evolution of architecture of TransF NN, attention mechanism, and hybridization with other methods, during these few years, revolutionized computational modelling paradigm. This made a niche in Data Science dealing with multi-modal data (viz.Text, numerical time-series, sound (speech), and image/video sequence) with local and global inter-dependencies.

Natural Language processing (NLP) : Some of earlier models in use were word2vector (2013), MLP, RecNN (1997-2015), capsuleNN, LSTM, GRU and Transformer (2017-). The advances in modules with LSTM are Bidirectional-LSTM, LSTM+attention, LSTM+seq2seq model, LSTM+Reinforced Lrning, LSTM+self sup Lrning, LSTM+Tranformer, peep- LSTM, and Hierarchical LSTM.

The language models (LMs) are also categorized as Large language models (LLMs), Small language models (SLMs) and Large/Small language models (LSLMs) based on training data size and number of parameters.

Transformer models: Attention and MLP NN are the two basic modules of a Transformer invoked in 2017 by Vaswani. Bahadaname (2014) and Luong (2015) invoked the concept of attention in pre-transformer era. The Transformer model with self-attention layers achieved state-of-the-art results in machine translation and completely replaced RNNs.

The frames proposed during this decadal period are Generative Pre-trained Transformer (GPT-x: x=1 to 4). BART (Bidirectional and Auto-Regressive Transformers), BERT (Bidirectional Encoder Representations from Transformers), T5 (Text-to-Text Transfer Transformer), PaLM (Pathways Language Model), CLIP (Contrastive Language-Image Pre-training), DALL·E (Multimodal models), BARD, and LLaMA (Large Language Model Meta AI)-2023.

The evolution of Tranformers led to LinFormer, Longformer and Performer with high end technical features and applications.

A few of Transformer neural nets (TransF-NN) or Transformer nets (TransF-N) architectures used in this state-of-knowledge-methods-module for dataToknowledge transformation are

- APT: Alarm Prediction Transformer,
- FastPCI, Swin Transformer Transformer,
- Cross-scale prototype learning transformer (Cplformer),
- Multiscale Network (MSNet),
- (Automatic Fusion Networks (AutoFuse),
- U-shape transformer,
- cross-wise transformer module (CTM),
- Transformer with Sliding-Window Dissimilarity Cross-Attention (SWDCA),
- Neural Networks To Vision Transformers (NN2ViT),
- Wavelet-domain Convolution (WeConv), Forest2Seq, RS-MOCO,
- Channel-wise Auto-Regressive entropy Model (WeChARM), Interview Training and Education Module (ITEM),
- RBMDC-Net,
- RISurConv: Rotation Invariant Surface,
- S-JEPA,
- REDIR: Refocus-free Event-based De-occlusion Image Reconstruction,
- DOLFIN: Diffusion Layout Transformers Without Autoencoder, FasterSTS:
- A Faster Spatio-Temporal Synchronous Graph Convolutional Networks,
- RP-Net: A Robust Polar Transformation Network,
- DMANet: Dual-Modality Alignment Network,
- CVT-Occ: Cost Volume Temporal Fusion for 3D Occupancy,

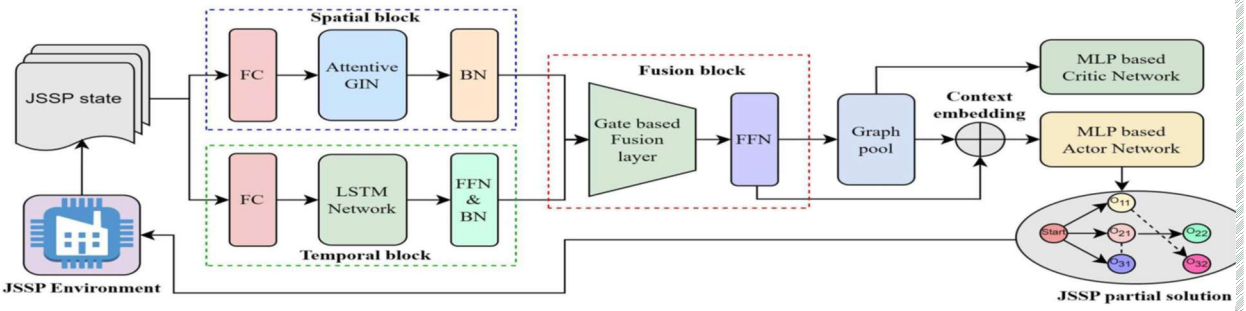
- Transformer network integrated into a chatbot interface,
- Swin transformers with the U-Net architecture incorporating residual blocks (RBs) and
- Attention mechanism and Trident Transformers (TT).

Keywords: Artificial intelligence (AI); Capsule Neural Nets— MLP-Attention Mechanism-TransFormer Nets—Hybrid TransFormer Networks--

CNN : [C [Computations; Computer; Chemistry, Cell, Cellestial, Cerebrum] NN [New News; News New; Neural Nets; Nature News; News of Nature;]]
Fits : [Figure Image Table Script;]

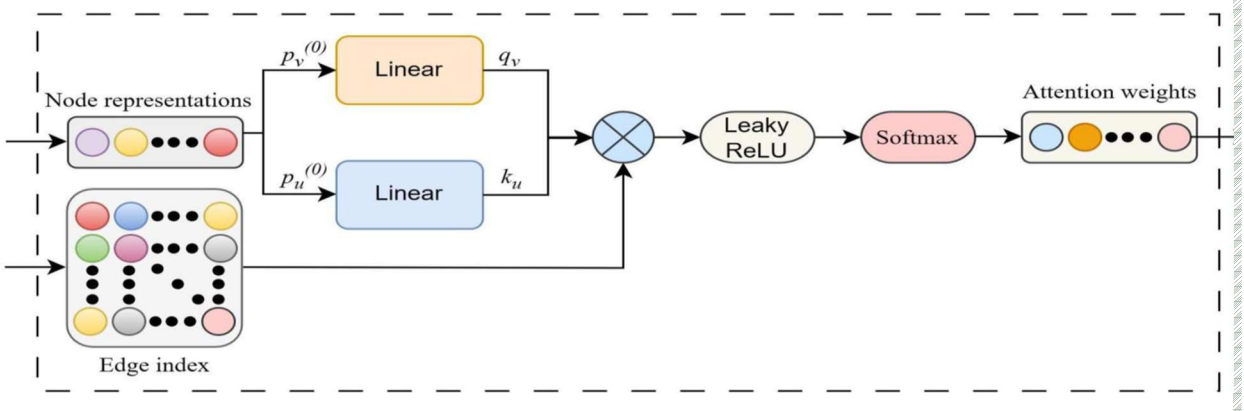
K(nowledge)Lab
rsr.chem1979

Transformer Net | **2025-53**



End-to-end architecture of DRL-STGNN for comprehensive scheduling

- ✓ Architecture comprises of
  - Spatial block (enclosed in a blue broken line)
  - Temporal block (enclosed in a green broken line)
  - Fusion block (enclosed in a red broken line)



Attention mechanism for aggregating neighborhood information within Attentive-GIN-based spatial block

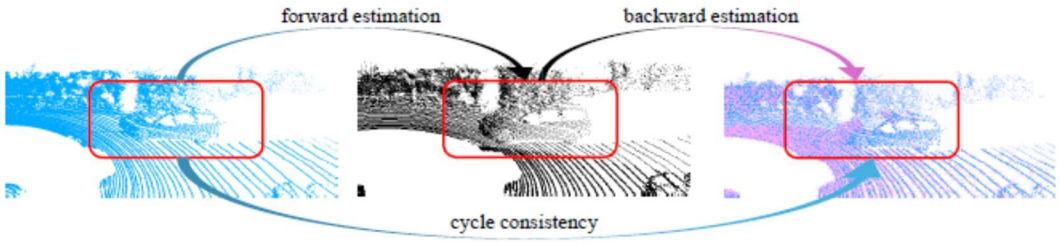


Fig. 2: (left) : frame t−1 ; (middle) : frame t ; (right) : the overlapping original frame t − 1 and the frame t − 1 estimated from frame t. FastPCI produces structure-aware motion and performs dual-direction motion estimation motivated by two facts:

- ✓ (1) structure consistency: an accurate motion preserves structure of objects, e.g. the car in red box;
- ✓ (2) cycle consistency: point cloud at frame t (right) is predicted from frame t−1 (left) by the estimated motion, meanwhile frame t−1 can be reconstructed from frame t (middle) by applying the reverse estimated motion (right).

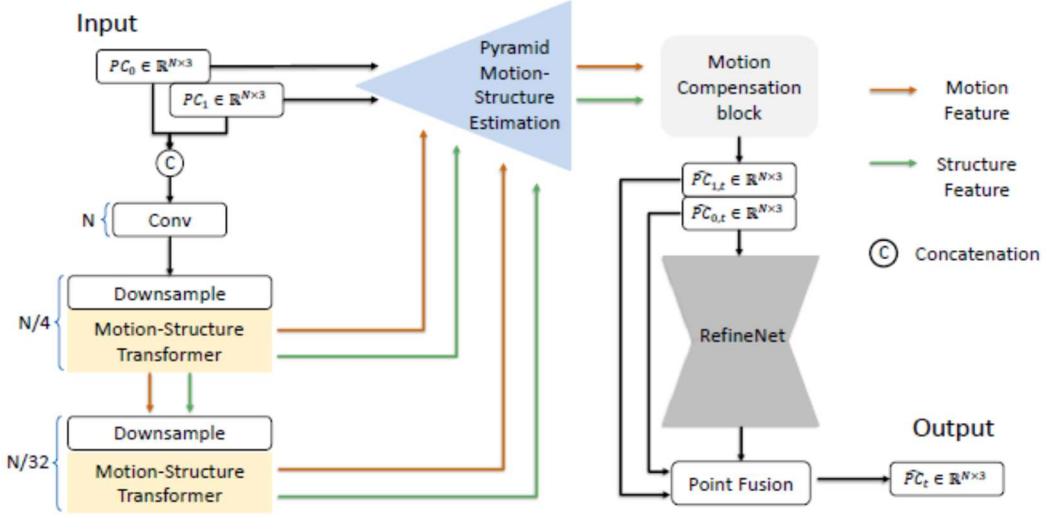
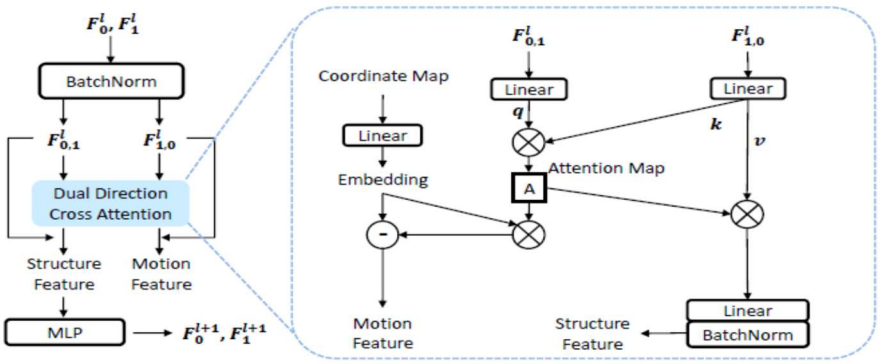


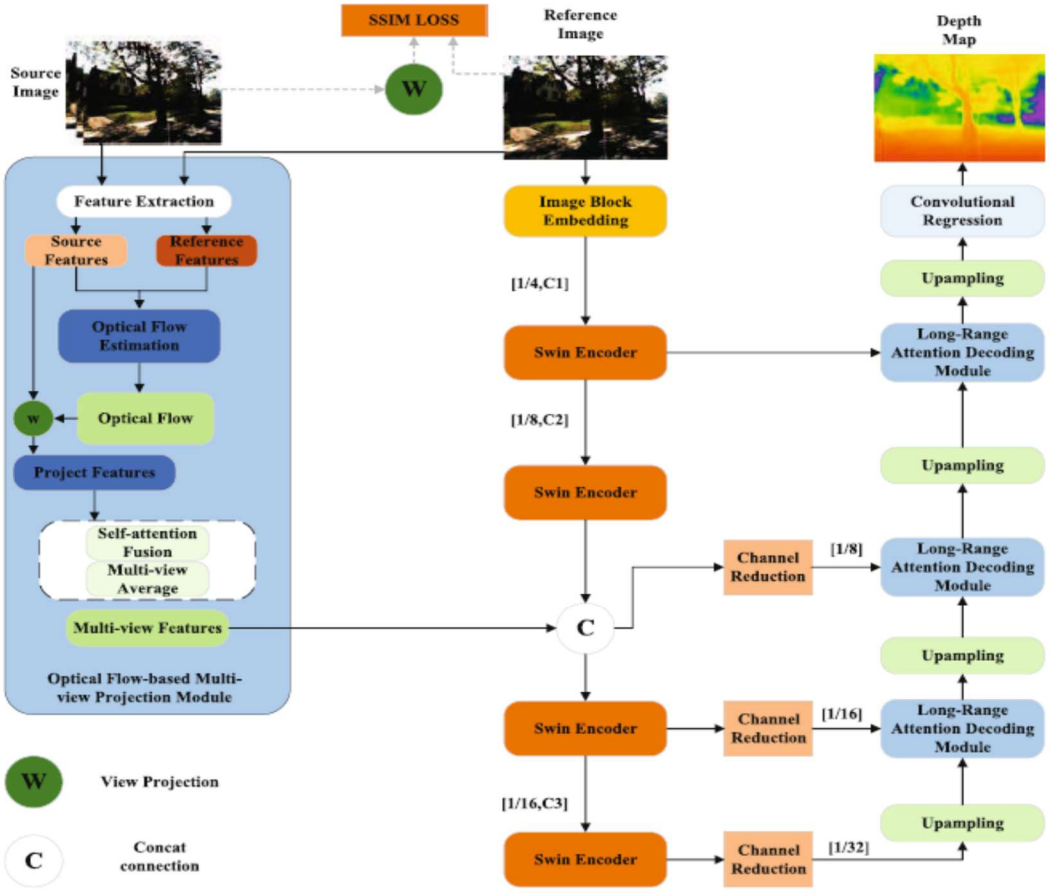
Fig. 3: Overview of FastPCI pipeline. Given two input frames $PC_0 \in \mathbb{R}^{N\times 3}$ and $PC_1 \in \mathbb{R}^{N\times 3}$, FastPCI estimates both motion and structure using a Pyramid Convolution-Transformer network. The estimated motion is used to warp the input frame to produce interpolated frames. RefineNet further refines the interpolated frames, and outputs the final frames from the fused forward and backward estimates.



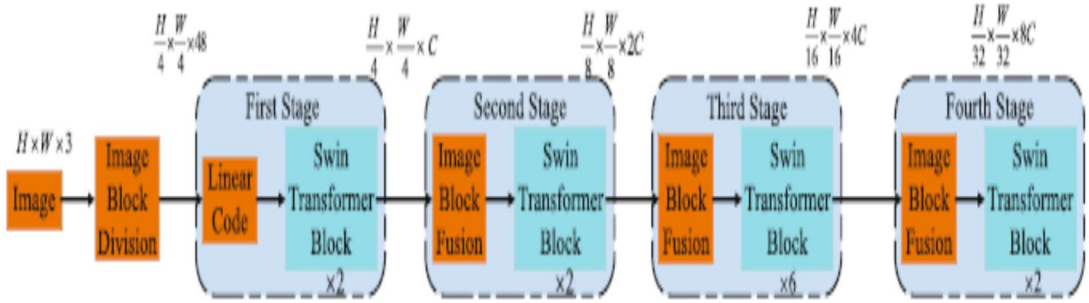
Motion-Structure Transformer.

- ⊗ and ⊖ denotematrix multiplication and element-wise subtraction
- Motion-Structure Transformer
  - takes a bidirectional point features input
  - perform a Dual-Direction Cross-Attention across forward and backward features
  - structure motion features are closely related to each other to learn a structure-aware motion.

Transformer Net	2025-55	

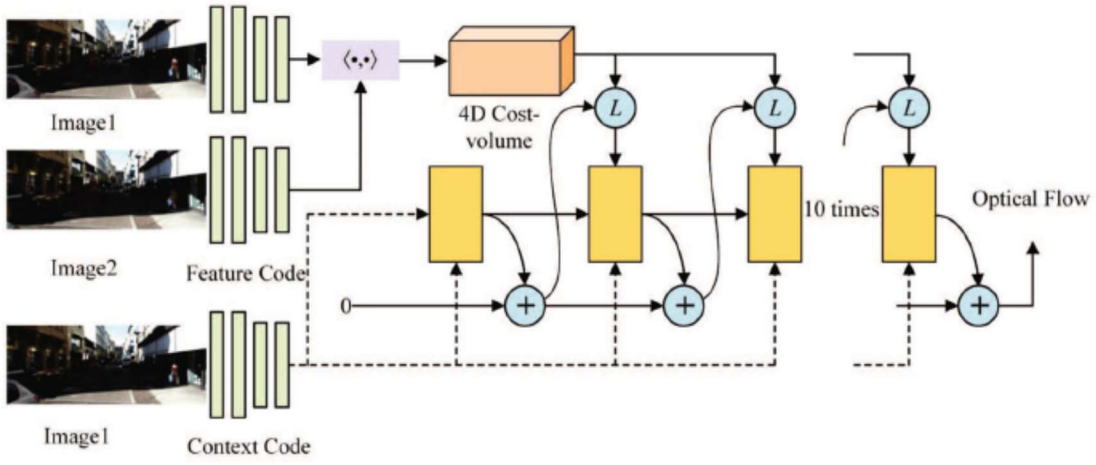


Structure of depth estimation algorithm assisted by multi-view projection



Swin Transformer-based coding structure

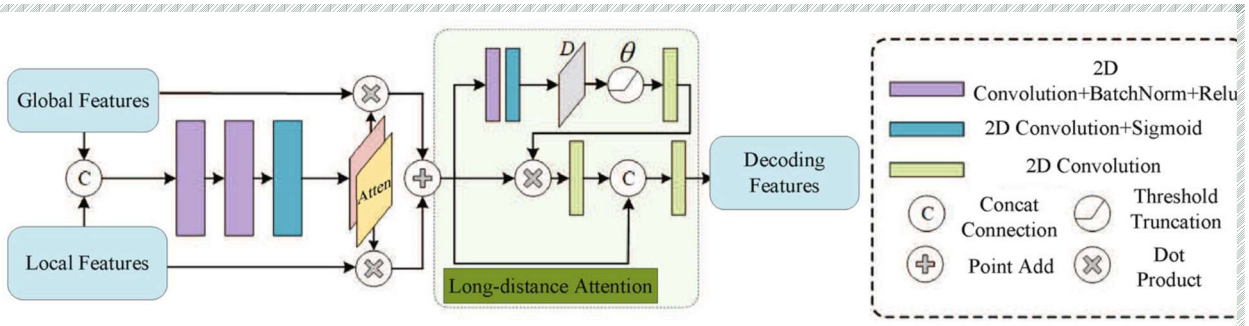
- ✓ Swin Transformer decomposes the entire encoder into several layers, each consisting of multiple small Transformer blocks
- ✓ It initially partitions image into several image blocks and encodes them into vectors
- ✓ These vectors are then passed through a linear coding layer to produce lowdimensional
- ✓ vectors, which are utilized in the subsequent hierarchical Transformer block
- ✓ Swin Transformer block comprises multiple grouped convolutional layers and several drift window self-attention modules



Structure of optical flow estimation network RAFT

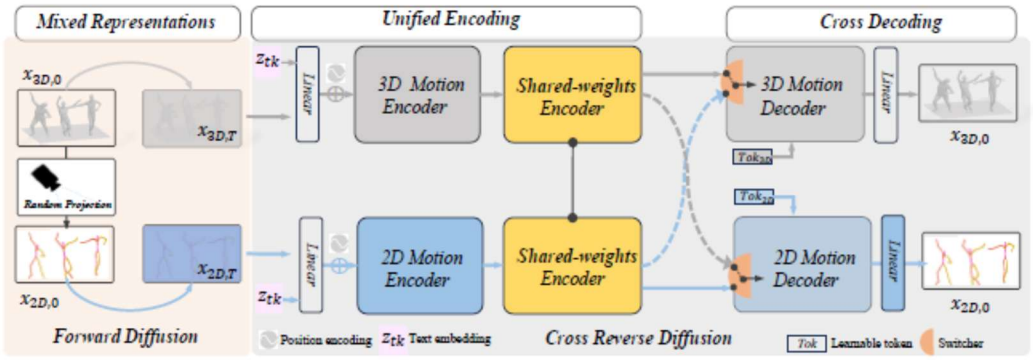
Reference View Projection View Source View

Multi-view projection effect based on optical flow



Schematic diagram
long distance attention decoder module

Transformer Net	2025-56	



Overview of CrossDiff framework for generating human motion from textual Descriptions

✓ Framework incorporates both 3D and 2D motion data, using unified encoding and cross-decoding components to process mixed representations obtained from random projection.

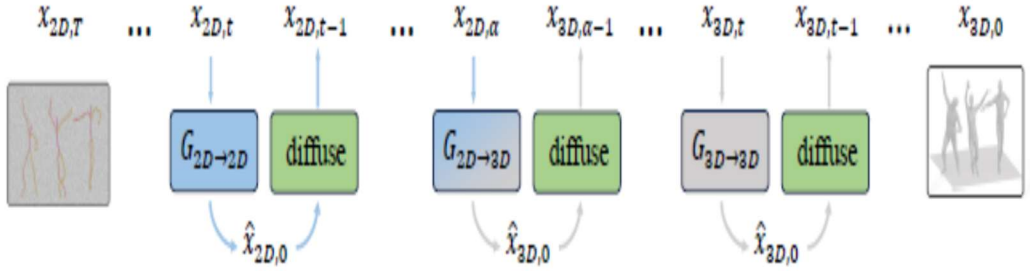
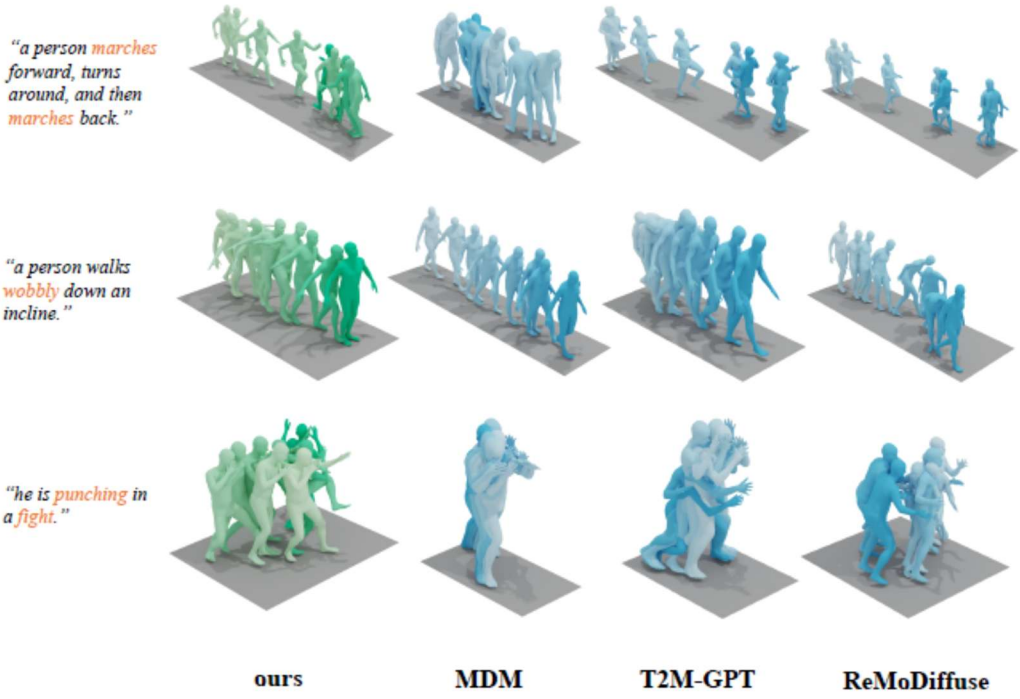


Fig. 3: Overview of Mixture Sampling

✓ **Original noise is sampled from a 2D gaussian distribution.**

✓ **From time-step T to α, crossdiff predicts the clean 2D motion x2d,0 and diffuses it back to x2d,t−1. In the remaining α steps, crossdiff denoises in the 3D domain and**

✓ **Finally obtains the clean 3D motion**



Qualitative results on HumanML3D dataset

✓ Compared with MDM, T2M-GPT and MLD

\+ Generated actions better convey the intended semantics

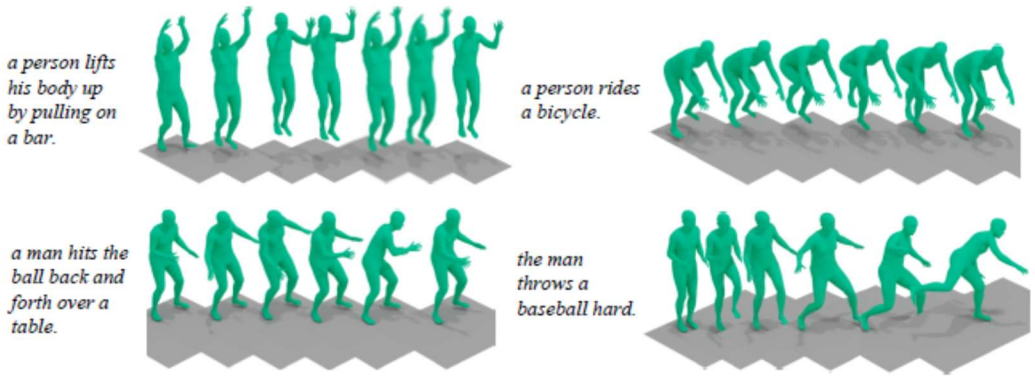
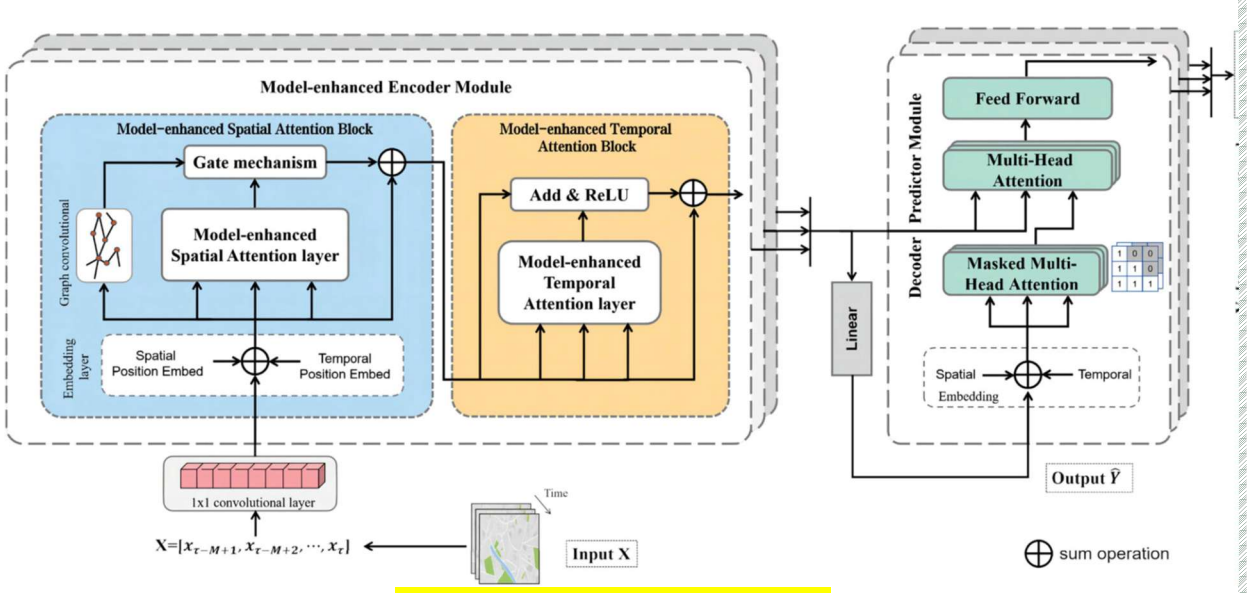
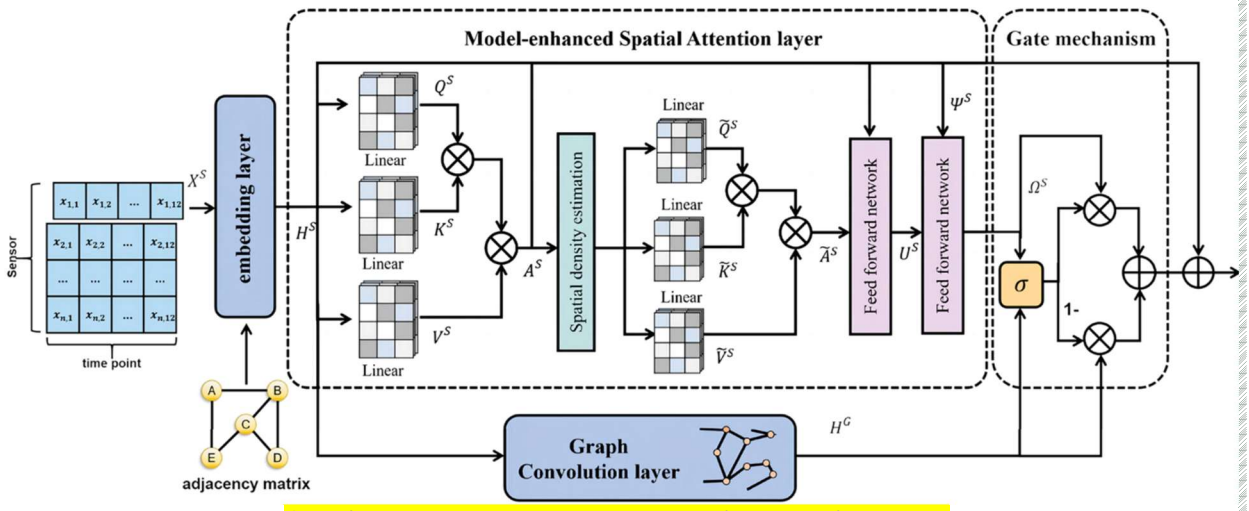


Fig. 6: Generating 3D movements without training on paired 3D motion and textual descriptions.



Architecture. MSTAN framework.

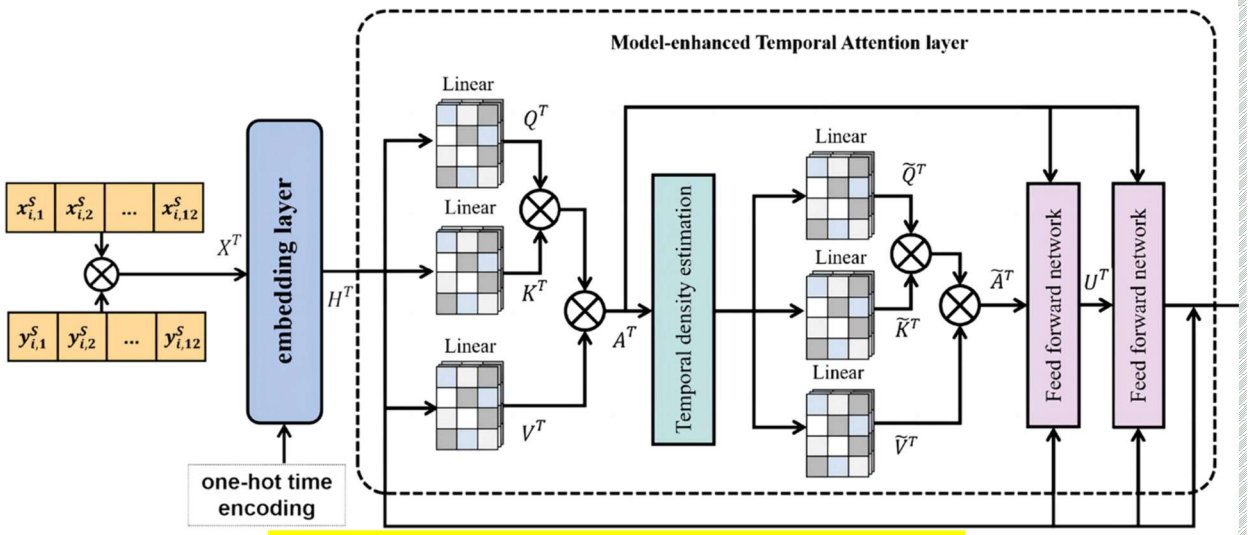
- Framework consists of two main components
  - Stacked model-enhanced encoder module and
  - Decoder predictor module.
- Each spatial-temporal transformer block consists of
  - a spatial transformer and
  - a temporal transformer



Architecture. model-enhanced spatial attention block

Comprising a

- Position embedding layer
- Graph convolutional layer
- Model-enhanced spatial attention layer
- Gate mechanism

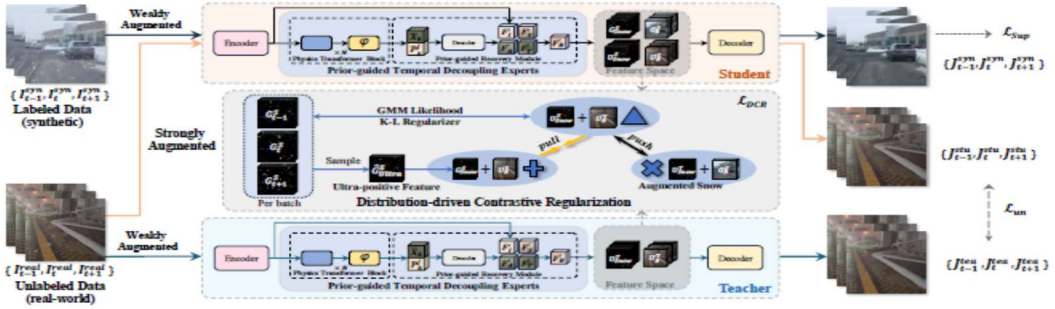


Architecture. model-enhanced temporal attention block

Modules

- ✓ Position embedding layer
- ✓ Model-enhanced temporal attention layer
- ✓ Information fusion feed-forward networks

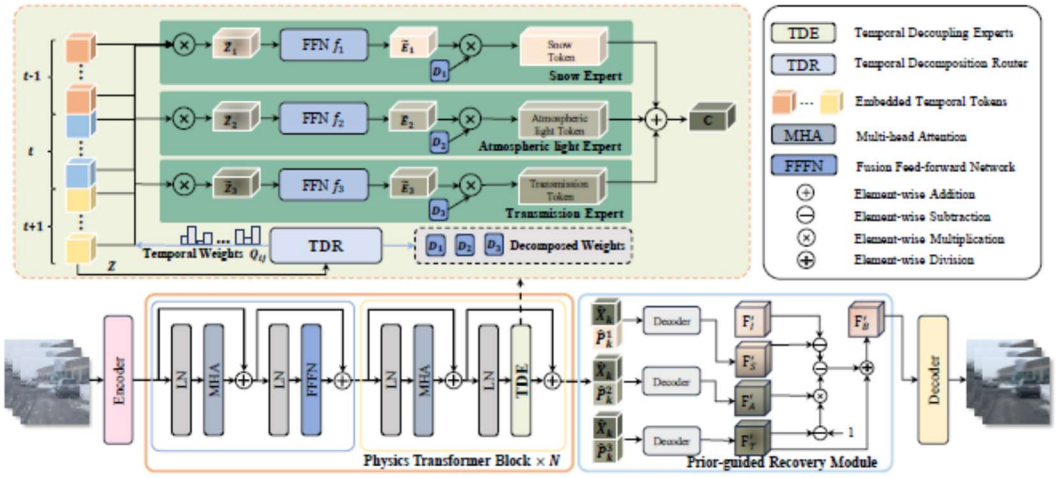
Transformer Net	**2025-57**	



Semi-Supervised Video Desnowing Network (SemiVDN)

SemiVDN

- ✓ Mean teacher scheme with a student model
- ✓ Teacher model
- ✓ Prior-guided Temporal Decoupling Experts
  - to decompose the physical components that make up a snow video in a temporal spirit
- ✓ After that compute supervised losses for labeled data and unsupervised losses for unlabeled data
- ✓ Based on the decomposed component features (F′ B and F ′ S) in representation space,
- ✓ Develop a Distribution-driven Contrastive Regularization to highlight the snow-invariant information by replacing the snow-specific feature in ultra-positive samples replacing the background in negative samples



Prior-guided Temporal Decoupling Experts framework.

- Input. snowy sequence encoded features
- Model. Physics Transformer Block (PTB)
- Module. Temporal Decoupling Experts module
  - To generate physics-specific components (i.e. S, A and T) for recovery.

- Temporal Decomposition Router

To compute the temporal weights Qij from the temporal dimension, which are subsequently employed to compute a linear combination of all input temporal tokens and Qij .

- Then each Expert (an MLP in this work) processes its temporal adaptive tokens to obtain the corresponding output component tokens. Finally, we employ the decomposed weights
- Temporal Decomposition Router
  - To convexly combine all the component tokens.
- Output combined features
- X k and physics-specific features pj k are subsequently input into the
  - Prior-guided recovery module decoder
    - To generate the ultimate desnowed results

Fig. 6: Samples of the proposed real-world video dataset for video snow removal.

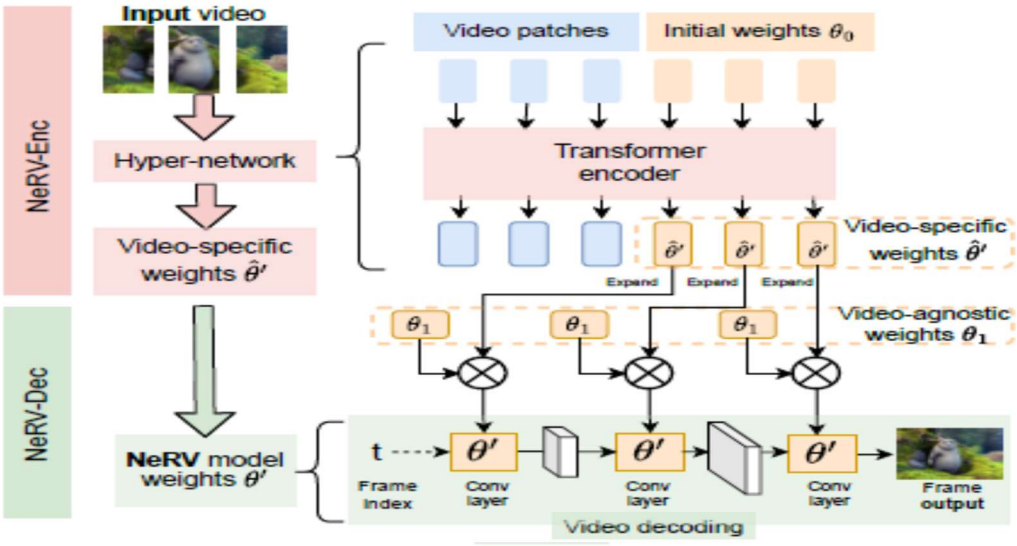
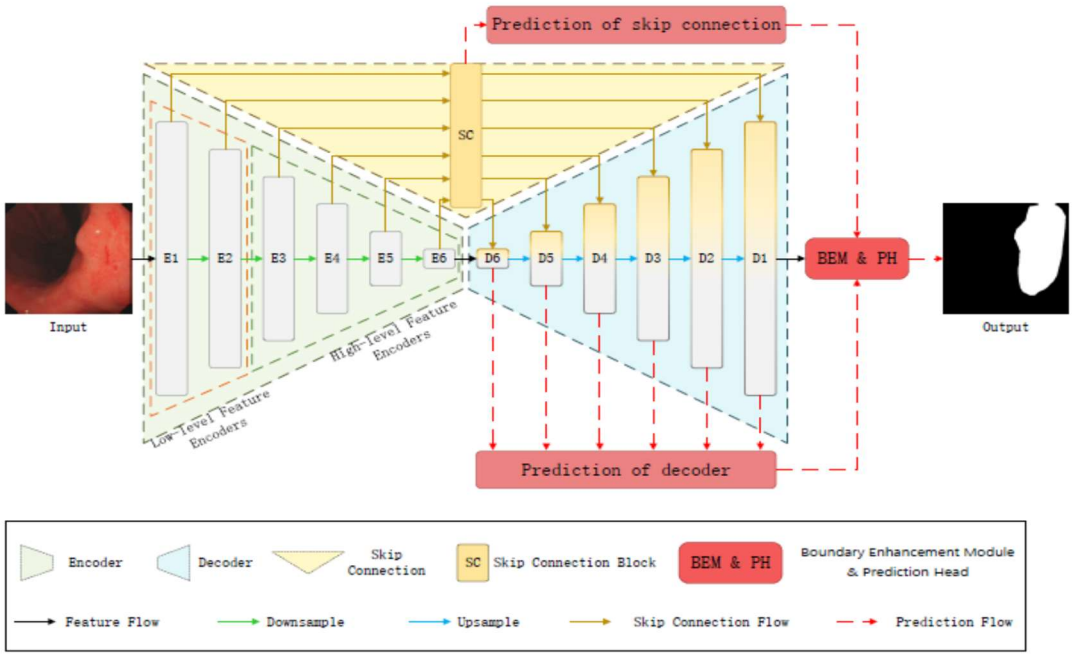
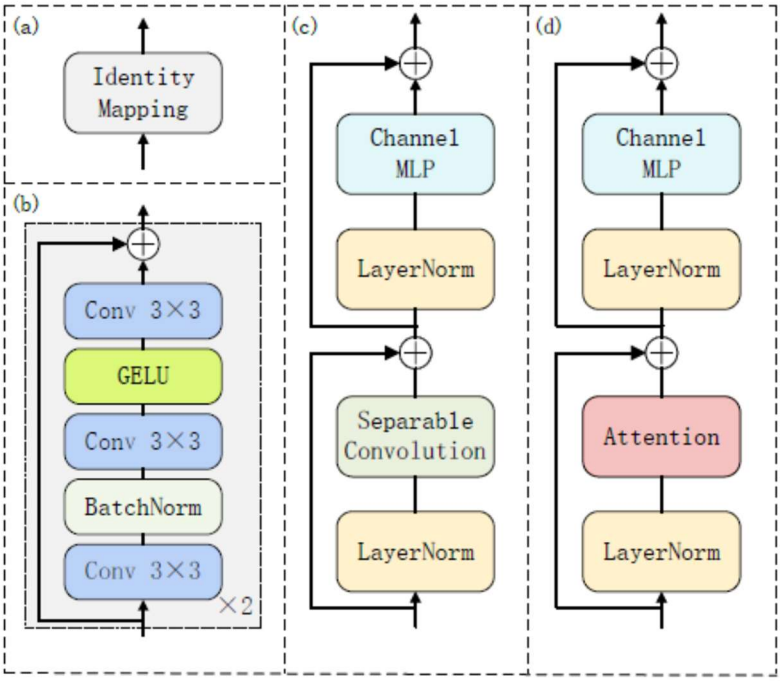


Fig. 2: Top: video encoding. NeRV-Enc processes the input video x to get video-specific weights $\hat{\theta}'$ using the hyper-network. **Bottom:** video decoding. NeRV-Dec generates final NeRV weights θ' and reconstruct video $\hat{x}$.

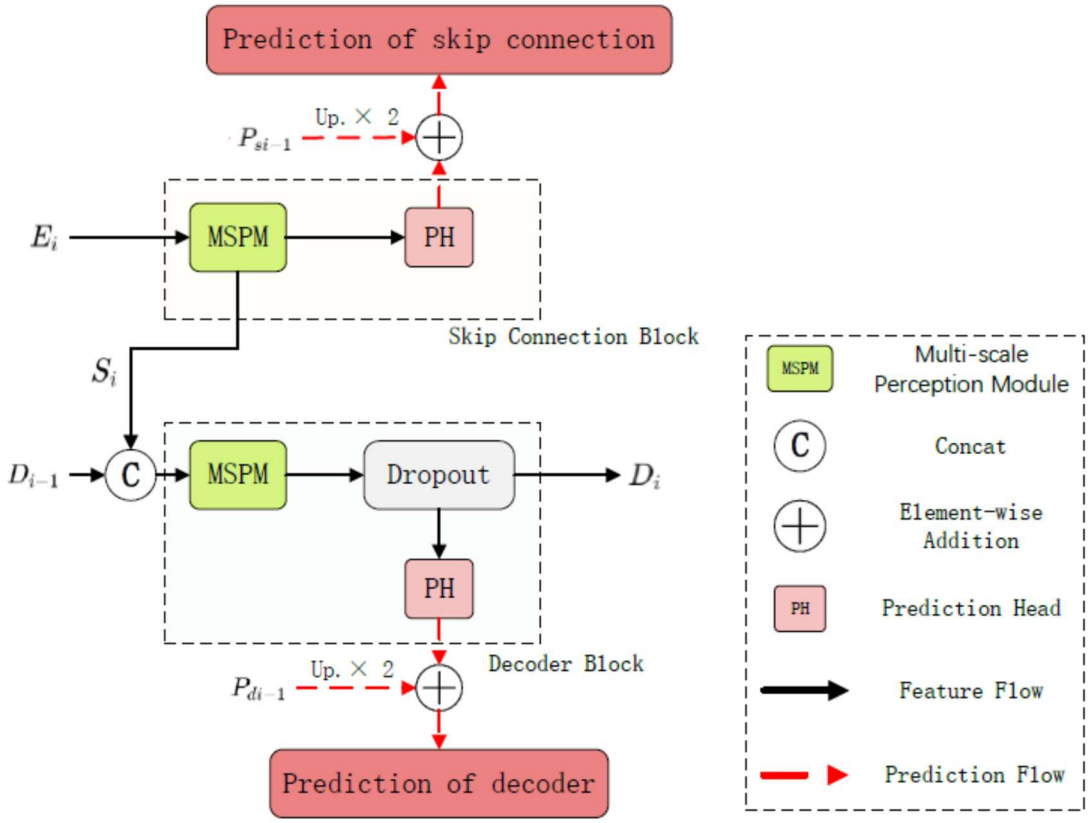


Framework of MSNet.

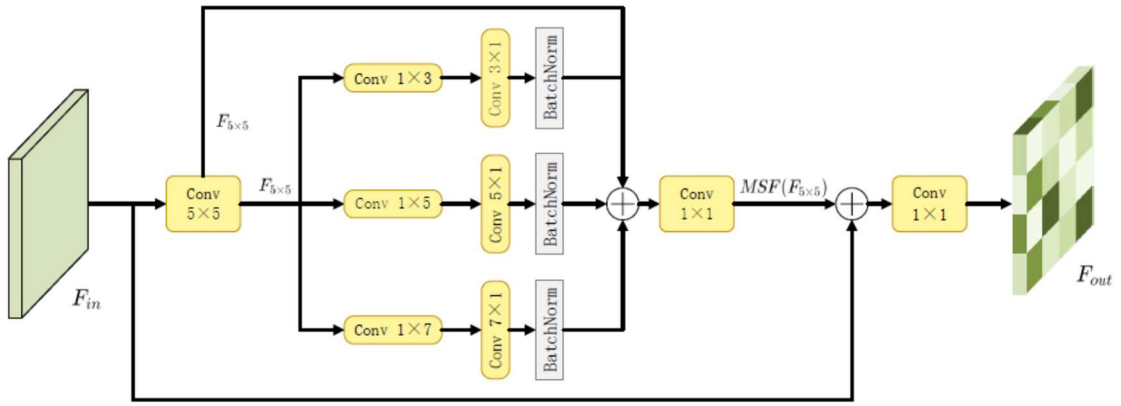


Hybrid encoder for the proposed method.

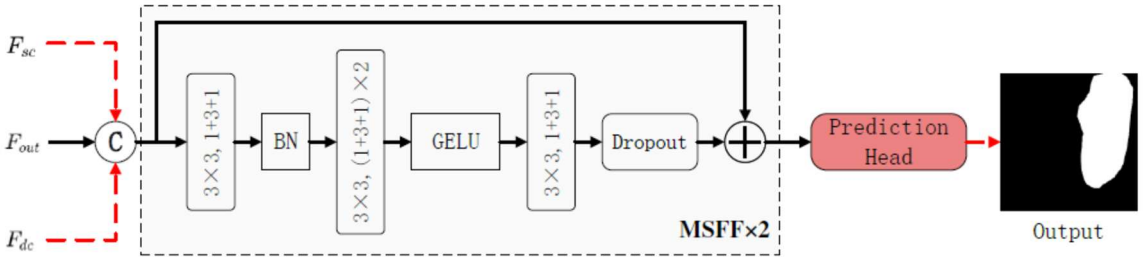
- **(a) and (b) represent the low-level feature encoders**
- **(c) and (d) represent the high-level feature encoders**



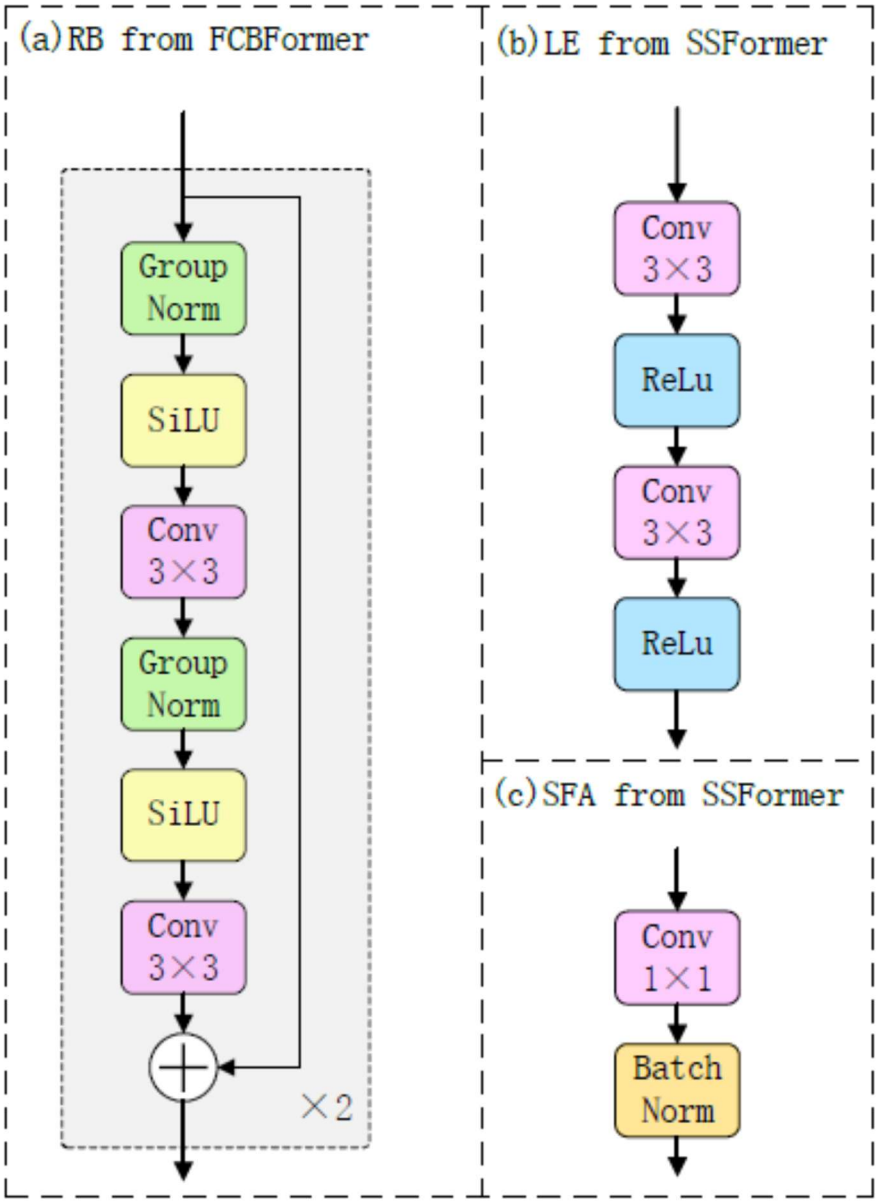
Details of Skip connection and decoder



Details of the proposed Multi-Scale Perception Module (MSPM)

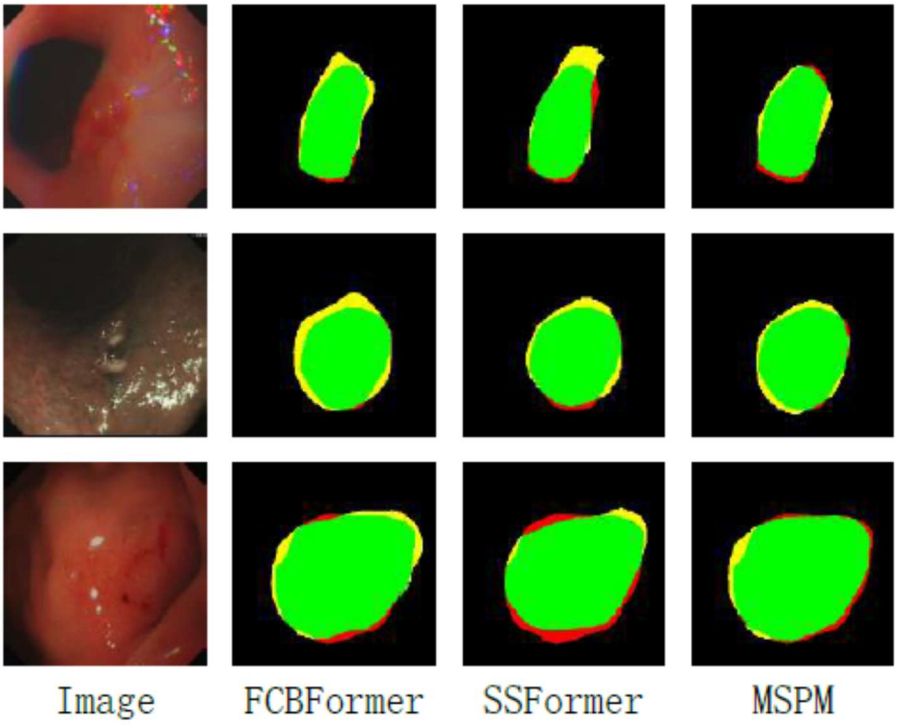


Details of the proposed Boundary Enhancement Module (BEM)



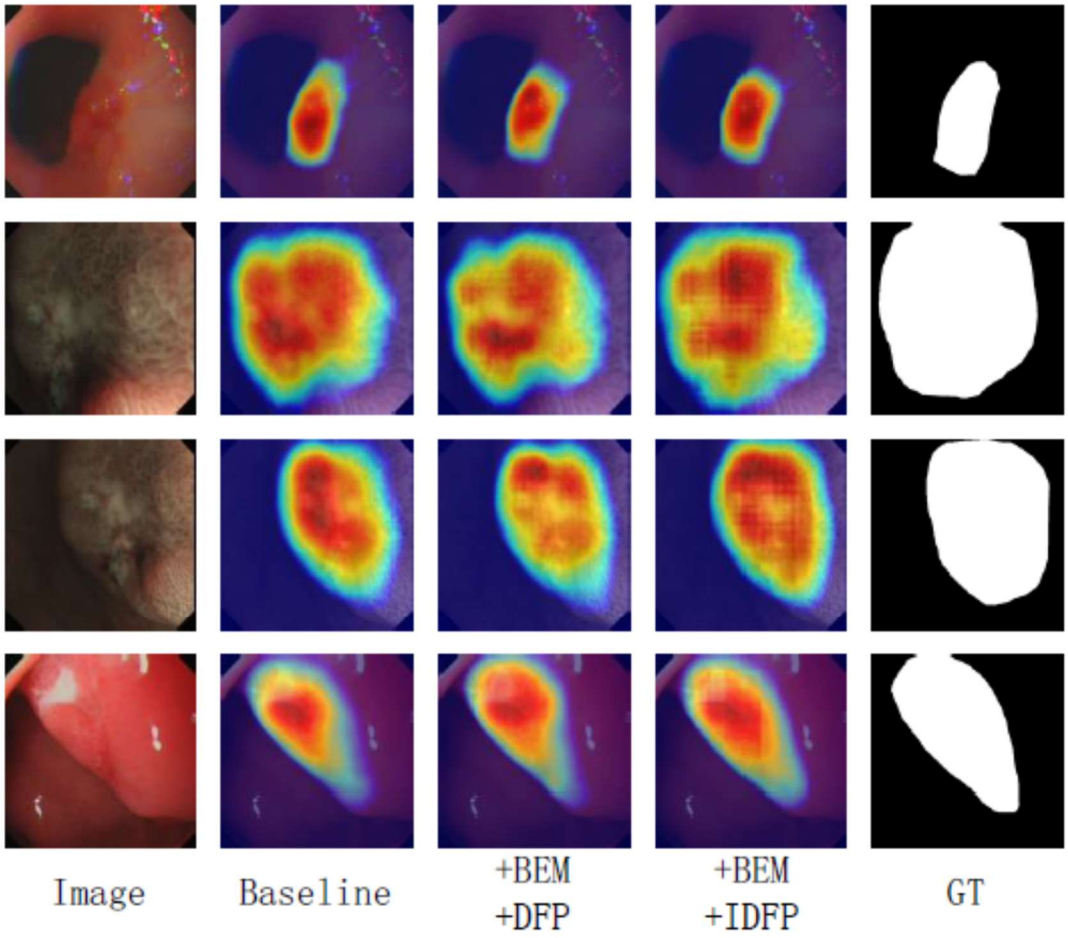
Using modules from different models in ablation experiments

- **(a) from the RB module in fcbformerfor skip connection and decoder**
- **(b) for the skip connection**
- **(c) for the decoder, from SSFormer**



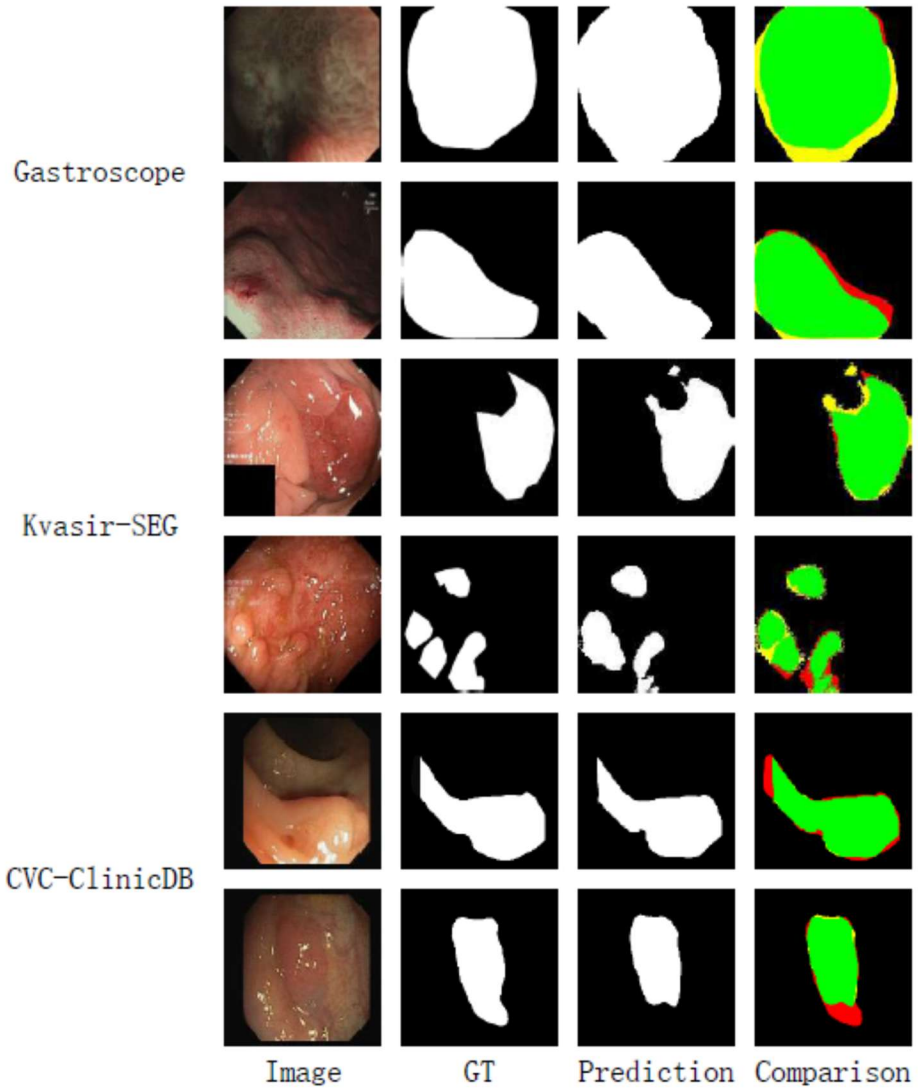
Visualization results of different variants on the gastroscopy dataset.

- ✓ Yellow part represents the prediction of the model
- ✓ Red part represents the true label
- ✓ Overlap between the two is green



Grad-CAM heatmap

✓ To visualize the influence of BEM and IDFP on the segmentation results



Typical failure cases of the proposed method on various datasets.

- First two rows belong to the gastroscopy dataset
- Middle two rows belong to the kvasir-seg
- Last two rows belong to the CVC-ClinicDB.

Transformer Net		**2025-60**	

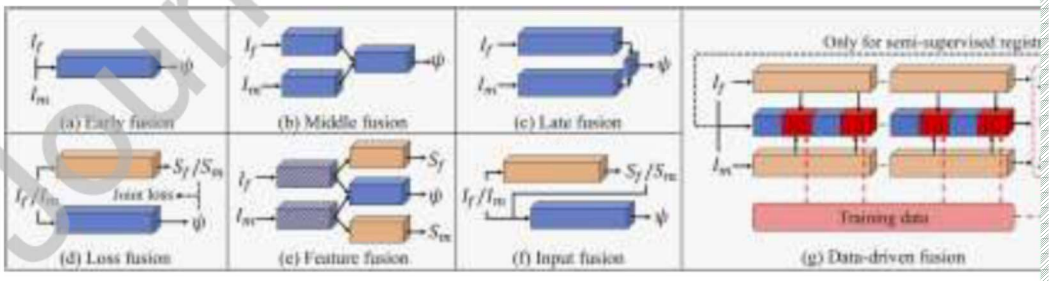


Fig. 1. Illustration of existing empirically-designed fusion strategies (a-f) and our data-driven f strategy (g) in unsupervised and semi-supervised settings. (a) Early fusion: I_f and I_m are concatenat input. (b) Middle fusion: I_f and I_m enter separate encoders with intermediate features fused. (c) fusion: I_f and I_m enter separate networks with resultant features fused. (d) Loss fusion: ψ and S_f/S mutually constrained by joint loss functions. (e) Feature fusion: multi-task networks are used with p feature shared. (f) Input fusion: S_f/S_m are fed as input for registration. (g) Data-driven fusion: t strategy is optimized during training. FG = Fusion Gate (FG) module; I_f = fixed image; I_m = m image; ψ = registration output; S_f/S_m = segmentation output for I_f/I_m (for semi-supervised registrati

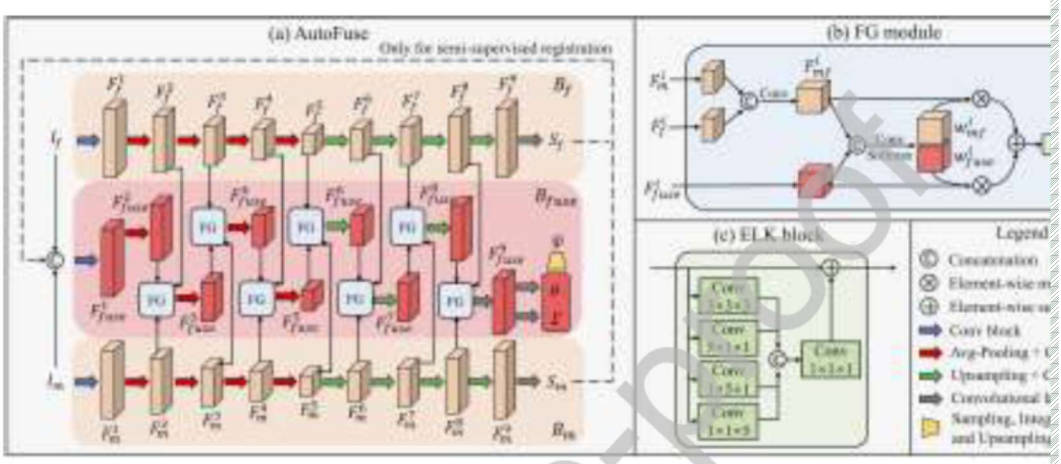
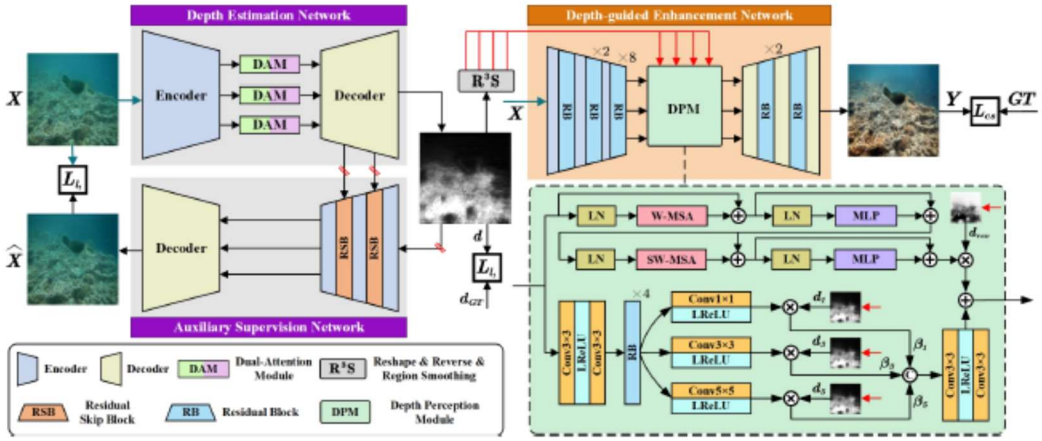


Fig. 2. Overview of the proposed Automatic Fusion network (AutoFuse)

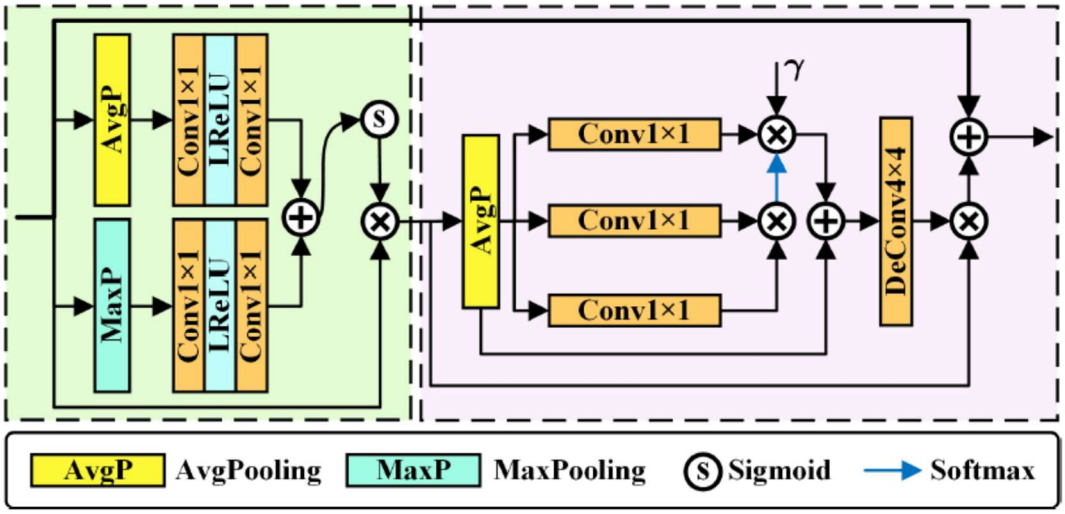
Architecture of
(a) AutoFuse
(b) Fusion Gate (FG) modules
(c) Efficient Large Kernel (ELK) blocks

- ✓ Skip connections of each branch are omitted in this figure for the sake of the clarity
- ✓ The branches Bm and Bf share the same weights



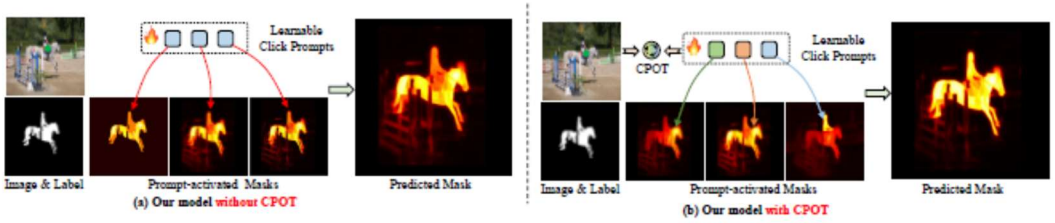
two-stage framework of UVZ, including DEN, DGEN, and ASN.

- All subnetworks
  - Adopt a standard encoder/decoder architecture, where the red slash indicates that ASN is only used for the training.
  - In the first stage, for a raw image X, DEN and ASN generate the depth map d and the regression image bX , respectively
  - For the second stage, with inputs X and d, DGEN generates the enhanced image Y



Structure of Dual-Attention Module (DAM)

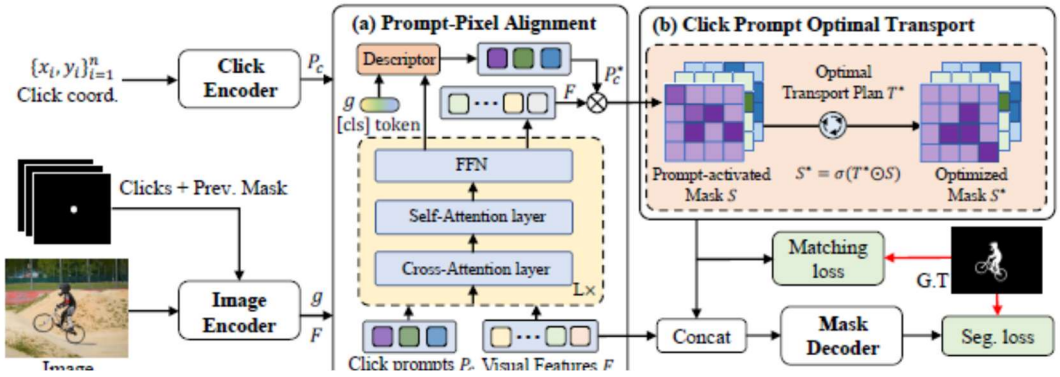
Transformer Net	2025-62	



Comparison between our model without and with Click Prompt

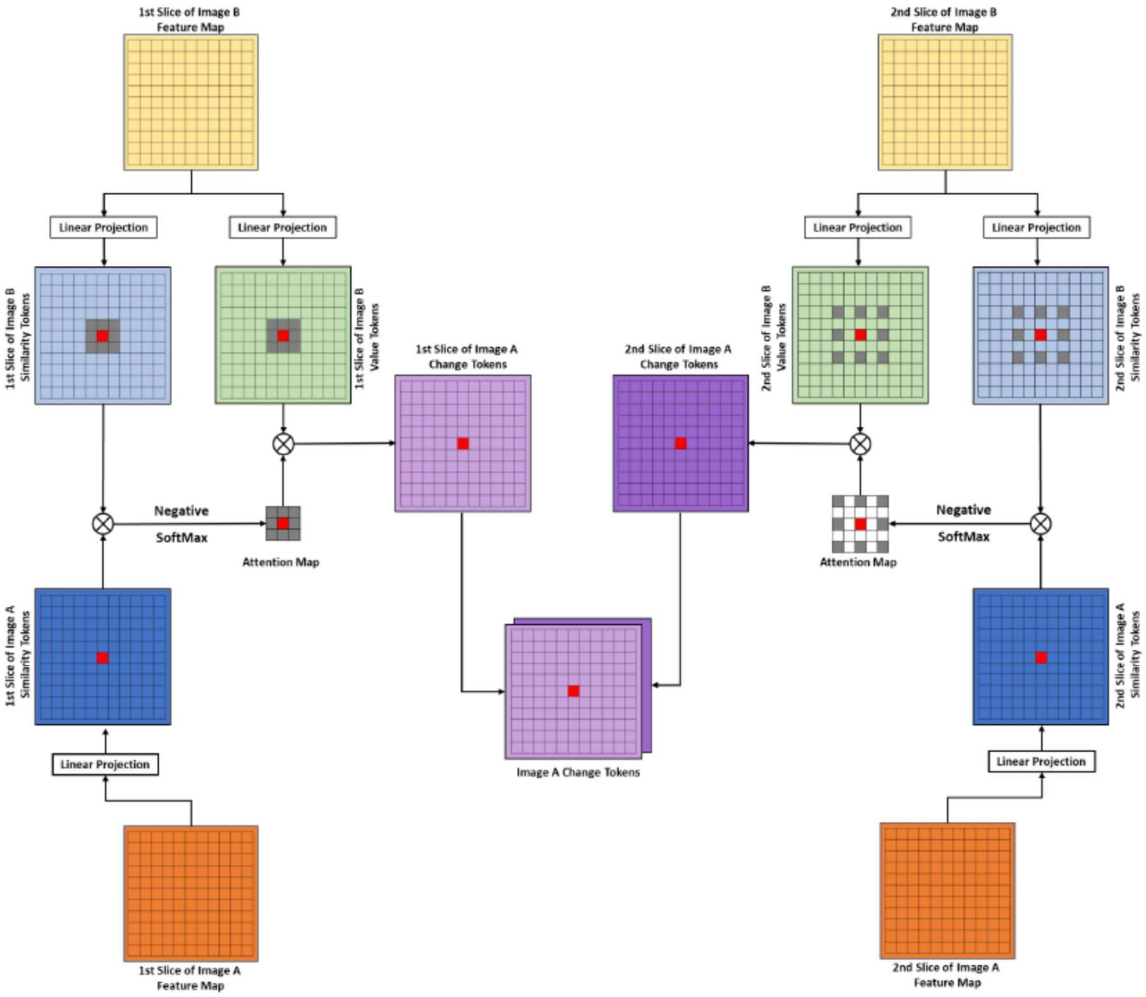
Optimal Transport (CPOT)

- Without CPOT, all click prompts tend to converge to one point, resulting in homogeneous prompt-activated masks and inferior mask prediction.
- (b) With the proposed CPOT, click prompts are encouraged to focus on distinct visual regions.
- Consequently, our model with CPOT predicts a more accurate mask by integrating diverse prompt-activated masks.



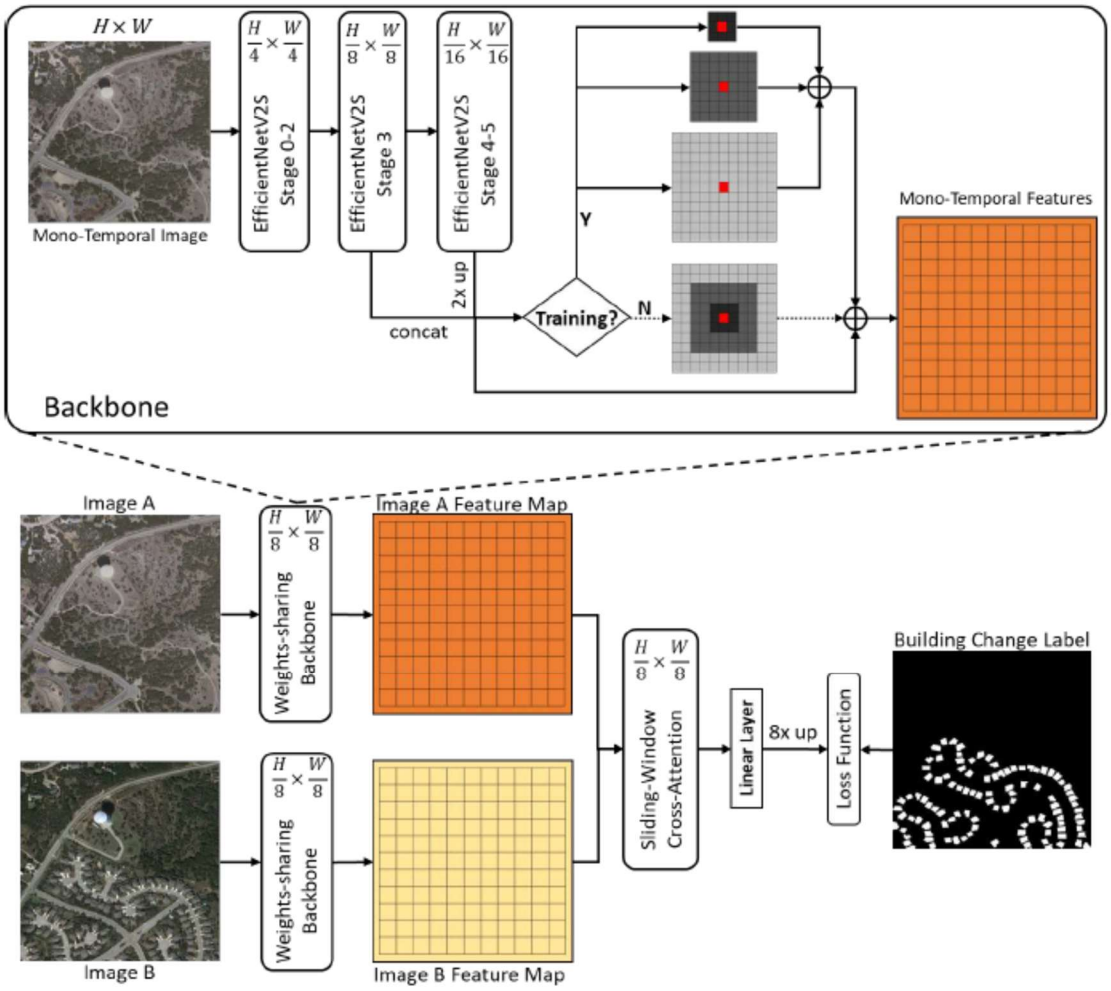
Overview of proposed Click Prompt Learning with Optimal Transport (CPlot)

- ✓ Given input image, click disk maps, and previous mask, the Image Encoder extracts visual features F.
- ✓ The Click Encoder initializes click prompts Pc with click coordinates.
- (a) Prompt-Pixel Alignment aims to align click prompts Pc with the visual features F in the feature space.
- (b) Click Prompt Optimal Transport adopts optimal transport plan to generate optimized mask S∗ from vanilla prompt-activated mask S. A lightweight mask decoder is used to implicitly analyze optimized prompt-activated mask with visual features and make mask predictions



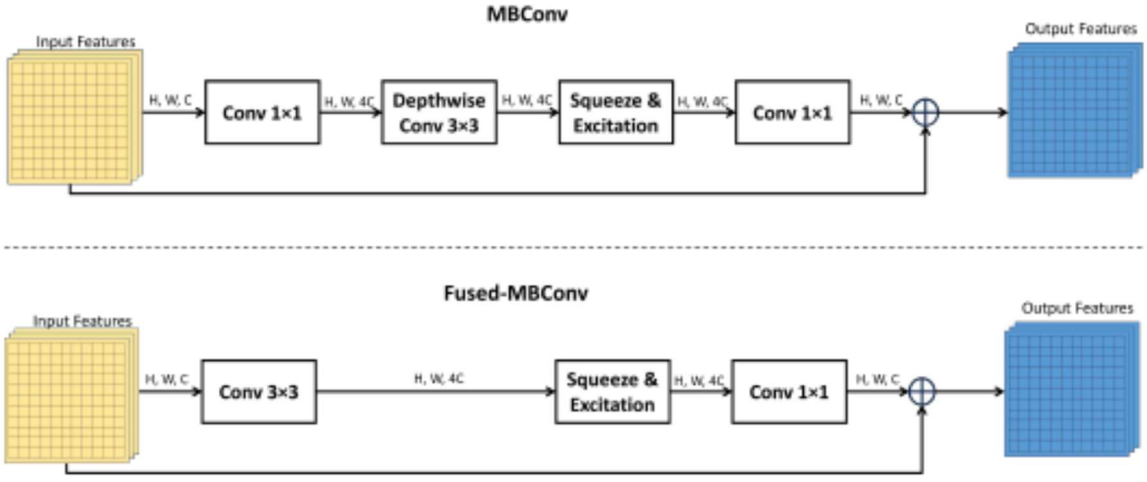
structure of sliding-window dissimilarity cross-attention module

- (A, B) ∈ {(1, 2), (2, 1)} denote two time points
- ⊗ represents matrix multiplication



Architecture. efficient change detection network

⊕ : Elementwise addition.



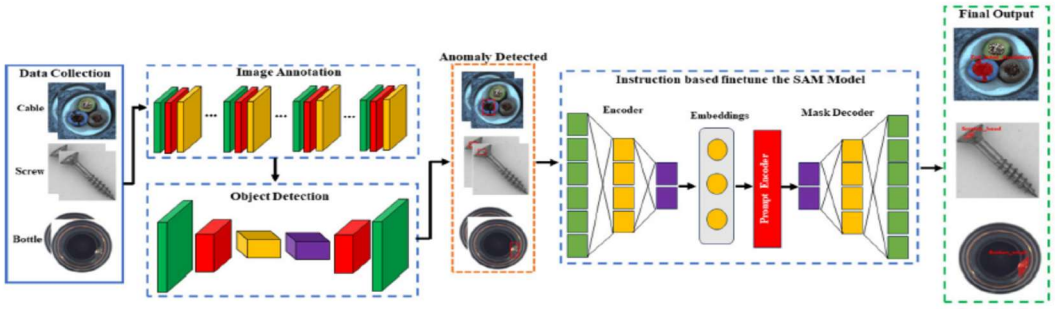
Structures. MBConv and Fused-MBConv

Table 2. Comparison of network parameters and computational complexity.

Method	#Param (M)	MACs (G)
STANet-BAM(ResNet18) [4]	12.2	49.2
STANet-PAM(ResNet18) [4]	12.2	50.2
DTCDSCN(SE-Res34) [7]	41.1	60.9
L-Unet [31]	8.5	
CDNet [32]	14.3	
MSCANet [33]	16.4	
BiT(ResNet18) [22]	3.0	35.0
SNUNet [14]	3.0	46.9
ChangeFormer(MiT-b1) [21]	13.9	26.4
IFN(VGG-16) [9]	36.0	316.5
FHD [34]	11.8	
ChangeStar(MiT-b1) [35]	18.4	33.7
Xu et al. [11]	61.4	
ChangerEx(ResNet18) [18]	11.4	23.9
ChangeStar(ResNet18) [35]	16.4	32.7
CDNeXt [36]	39.4	31.5
TransUNetCD [23]	95.5	
BAT [27]	6.9	40.3
SWDCA Network	5.4	25.0

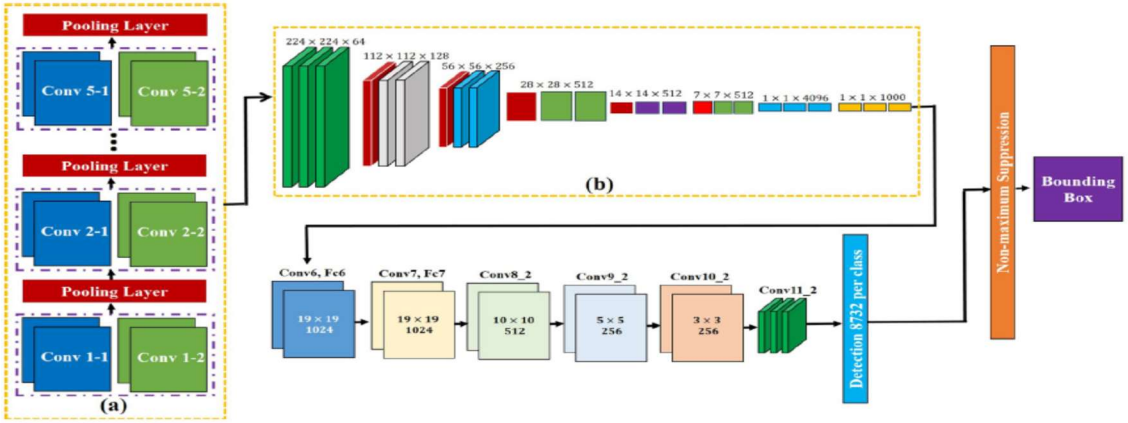
The computational complexity, quantified by multiply–accumulate operations (MACs), was evaluated using bi-temporal image pairs with a resolution of 512 × 512 pixels. The optimal value is indicated in red font, whereas the second-best value is represented in blue font. # means the number of.

Transformer Net	2025-64	

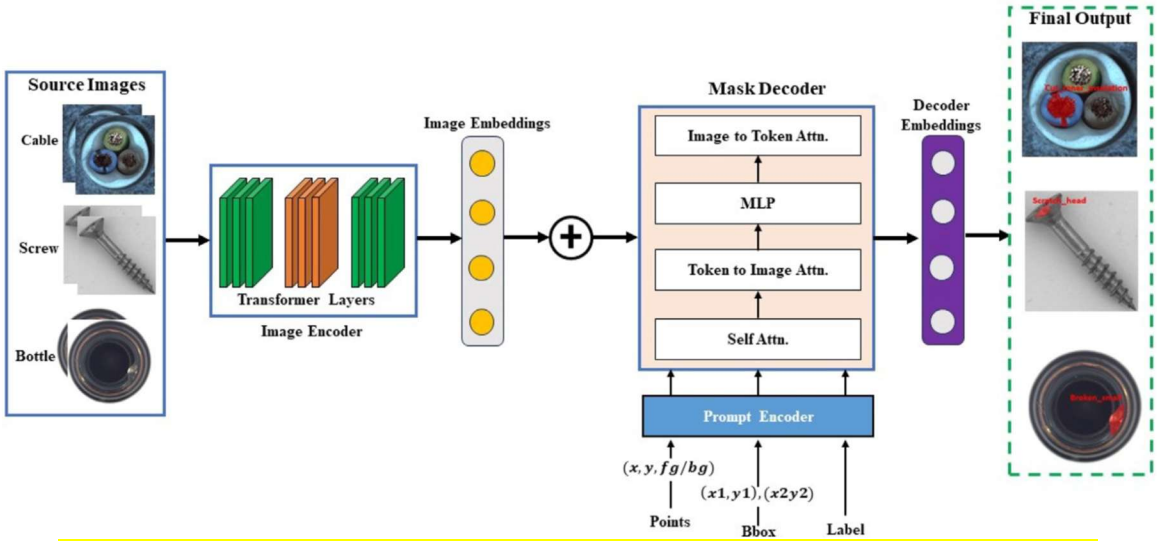


Framework. NN2VIT

✓ **For visual anomaly detection**

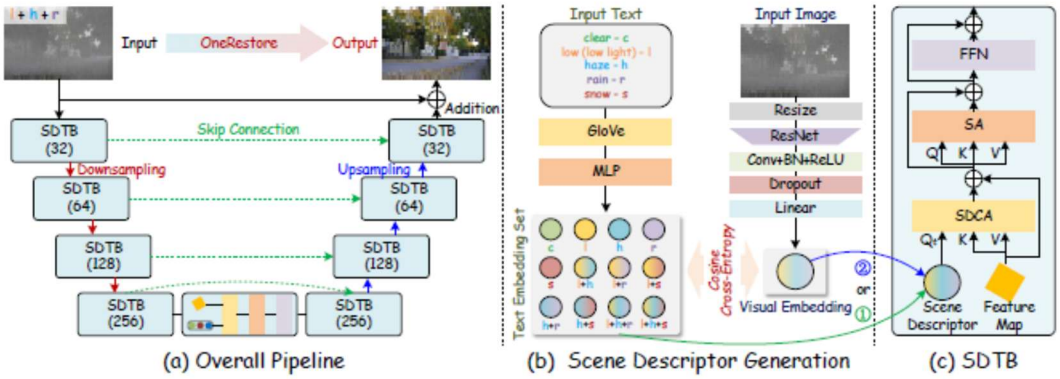


Single Shot multibox detector architecture used for Anomaly Detection



Prompt based fine tuning architecture of the SAM used for Anomaly Segmentation

Transformer Net	2025-65	



Architecture. OneRestore

- ✓ (a) Overall pipeline, where 32, 64, 128, and 256 represent the number of channels.
- ✓ (b) Scene descriptor generation, where scene descriptors are fed into each
- ✓ (c) Scene Descriptor-guided Transformer Block (SDTB) by manual text embeddings or automatic extraction based on visual attributes

Fig. 6: Comparison of image restoration on *low+haze+rain* (top) and *low+haze+snow* (bottom) synthetic samples.

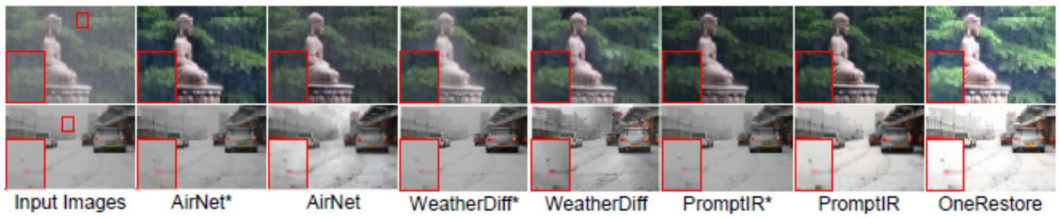


Fig. 7: Comparison of image restoration on *low+haze+rain* (top) and *low+haze+snow* (bottom) samples in real-world scenarios. * represents the utilization of original weights published in the author's code.

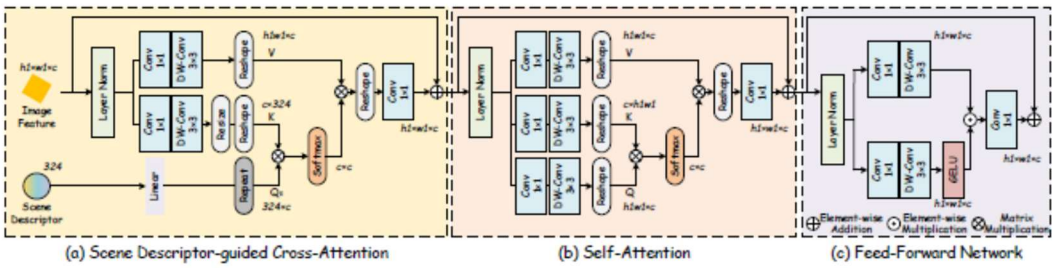
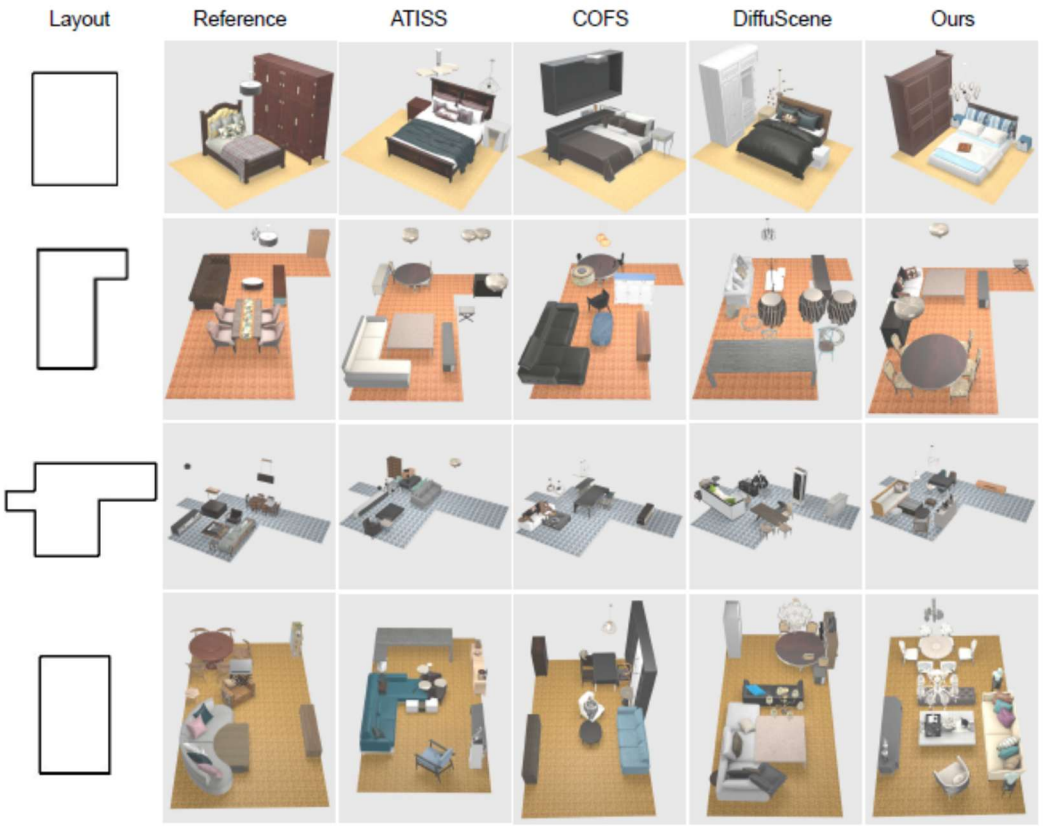
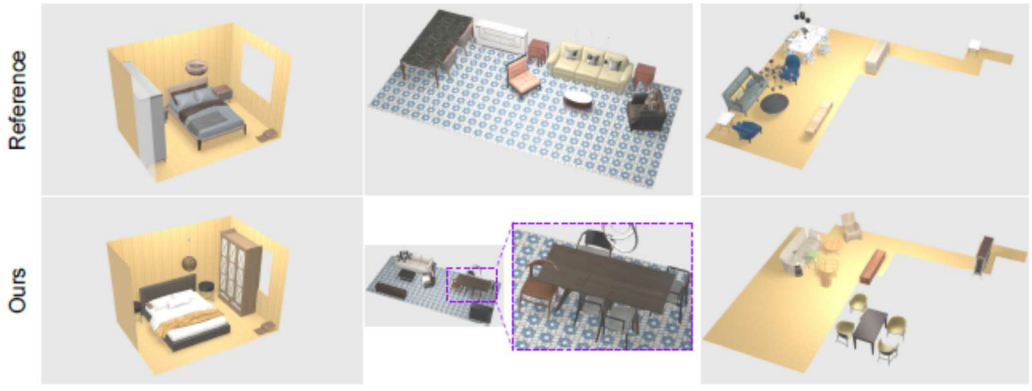


Fig. 11: Architecture of proposed Scene Descriptor-guided Transformer Block (SDTB).



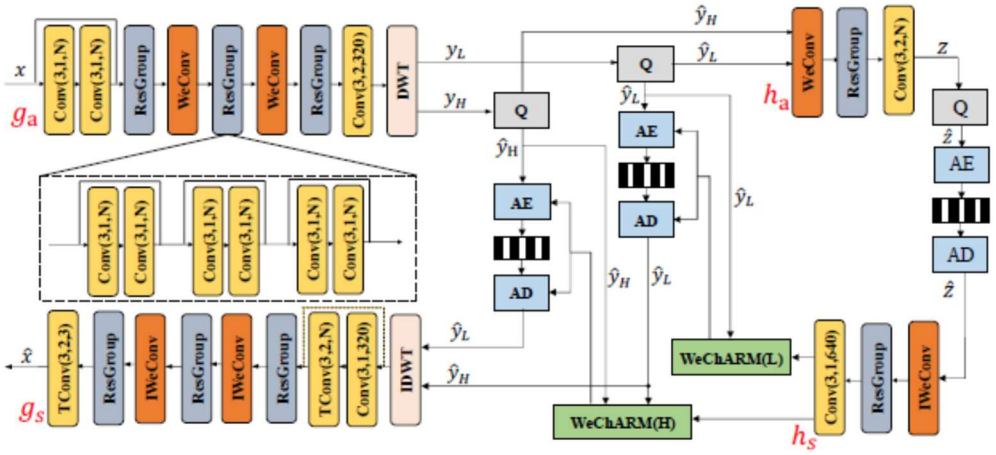
Qualitative comparison with the-state-of-the-art methods on scene synthesis

- Type of scenes: bedrooms (1st row), living room (2nd and 3rd rows) dining room (4th row).
- Reference is the scene from dataset with the same floor plan.



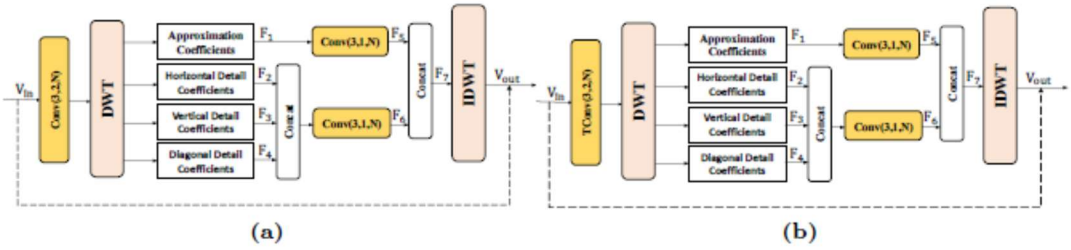
Failure cases

- ✓ Neglecting window placement (left); overlapping furniture arrangements (mid);
- ✓ Objects placed out of boundary in non-standard layouts (right).



Architecture. WeConvene scheme

- Conv(3, s, n) : a convolution layer with 3 × 3 kernel size, stride s, and n filters
- TConv(3, s, n) : transposed convolution.
- Dashed shortcut connections represent change of tensor size
- AE and AD: Arithmetic Encoder and Arithmetic Decoder



- **(a) Architecture. forward WeConv network with down-sampling**
- **(b) Architecture. inverse WeConv (IWeConv) network with up-sampling**

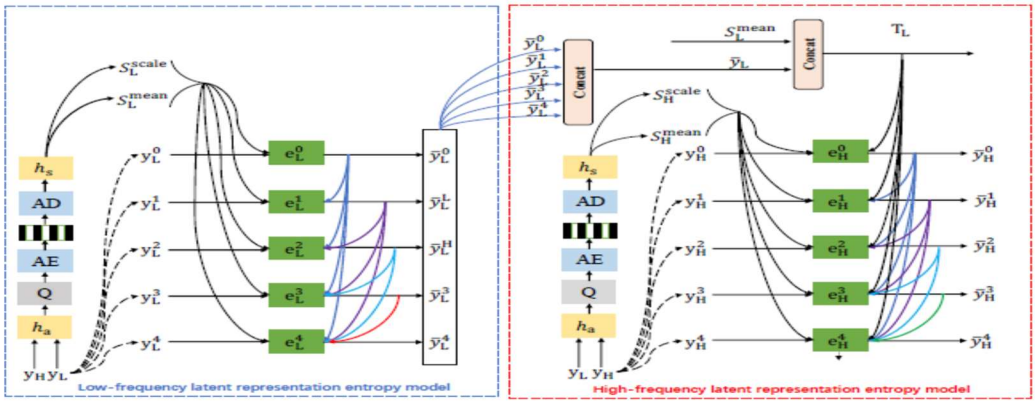
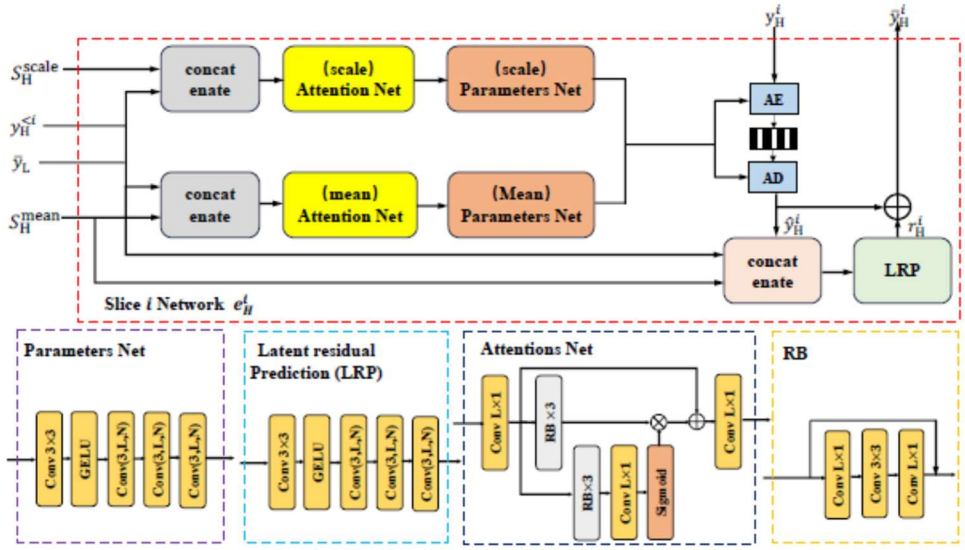
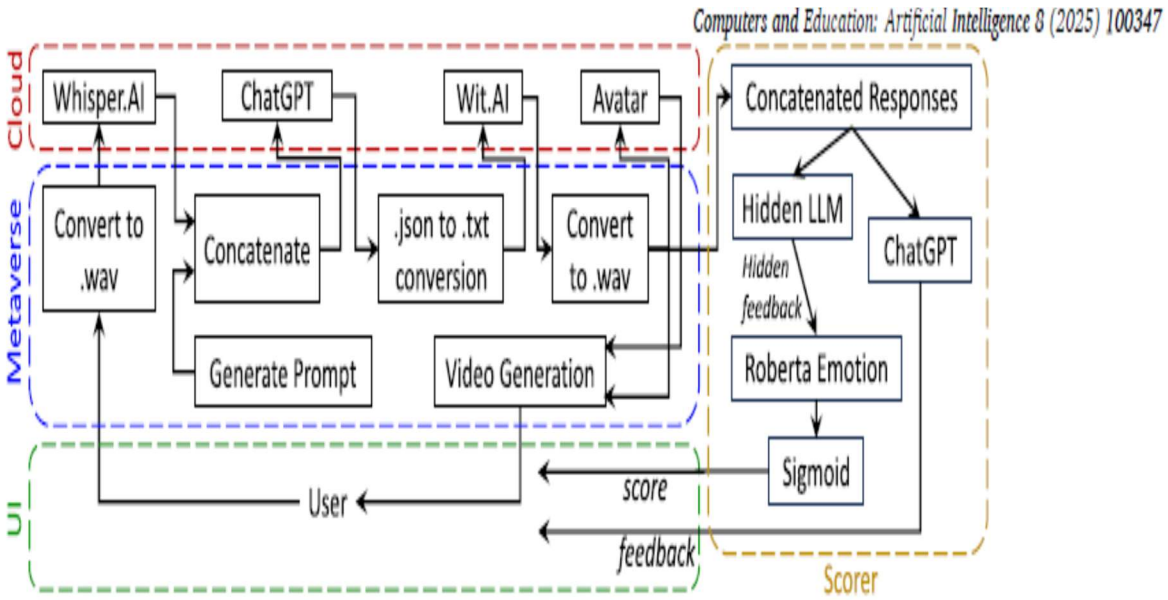


Fig. 4: The details of the proposed WeChARM modules for LF and HF subbands.



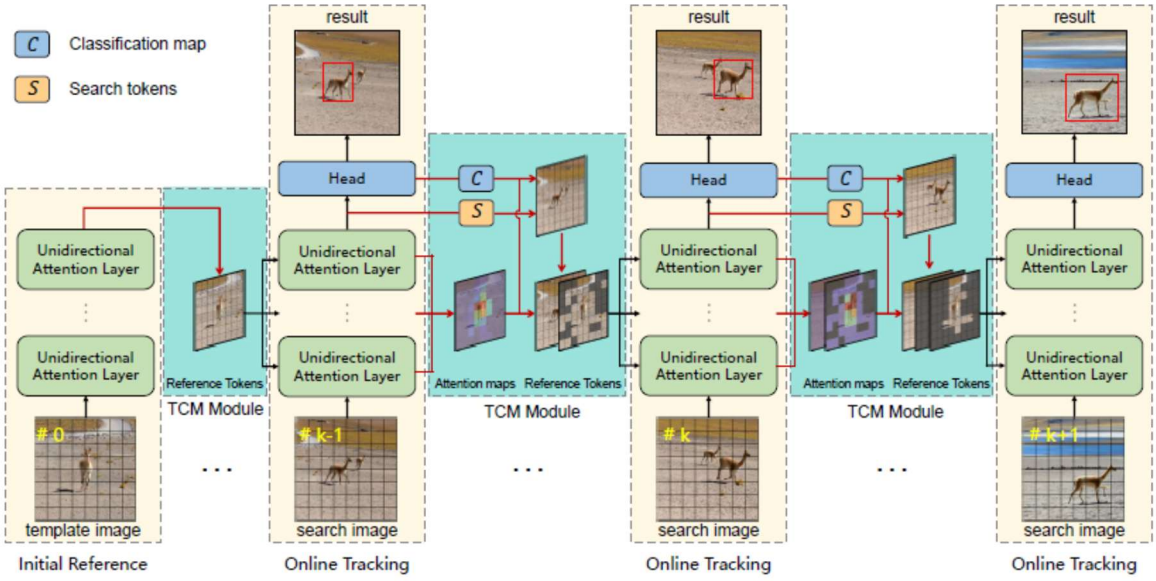
Slice coding network eiH for the HF entropy coding

Transformer Net	2025-68	



Three layers of ITEM

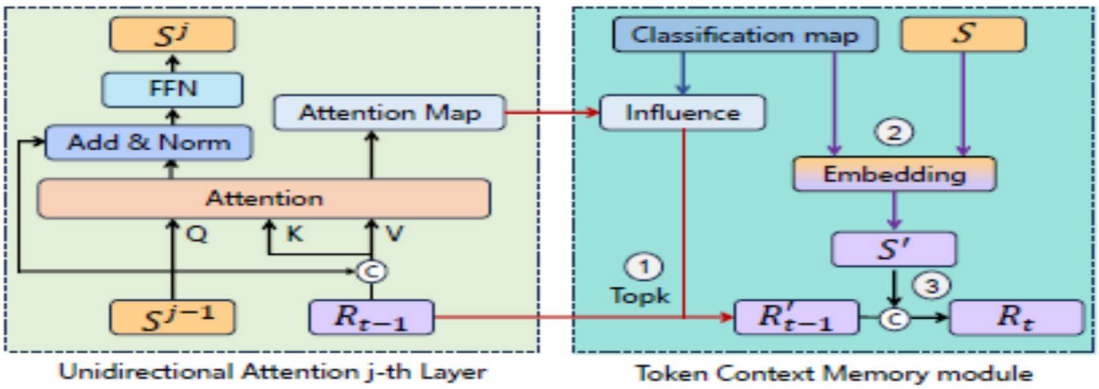
- The cloud layer hosts AI models from third-party service providers;
- Metaverse layer allows effective communication between the student and the AI models
- While the user interface layer makes the interview simulation realistic, immersive, and easy to use for students



Architecture. LMTrack

- ✓ LMTrack consists of three parts
  - a backbone with unidirectional attention,
  - a token context memory (TCM) module, and
  - a prediction head.

- Input of tracking pipeline contains a video frame and reference tokens being collected
- TCM module: utilizes classification maps and attention matrices to analyze the importance distribution of all reference tokens,
- hen collect the important reference tokens according to this distribution



- ✓ The unidirectional attention mechanism within the encoder layer is integrated with the token context memory module
- ✓ The inputs to the unidirectional attention include search tokens and reference tokens
- ✓ The token context memory module uses the attention map from unidirectional attention and predicted results to aggregate reference tokens

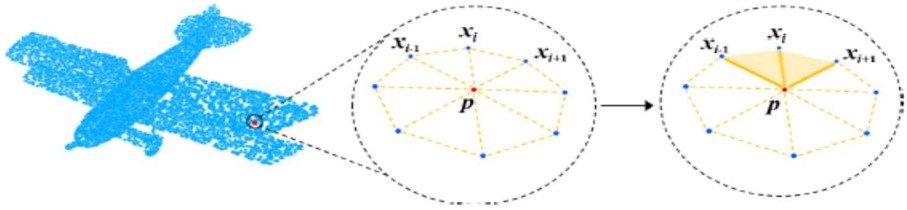


Fig. 1: Rotation Invariant Surface Property (RISP) construction: Given a point p as the reference point, K ($K = 8$ in this example) nearest points are selected (middle). For each neighbor x_i, two adjacent neighbors x_{i-1} and x_{i+1} are used to form two triangular local surfaces (right), based on which rotation invariant properties are constructed.

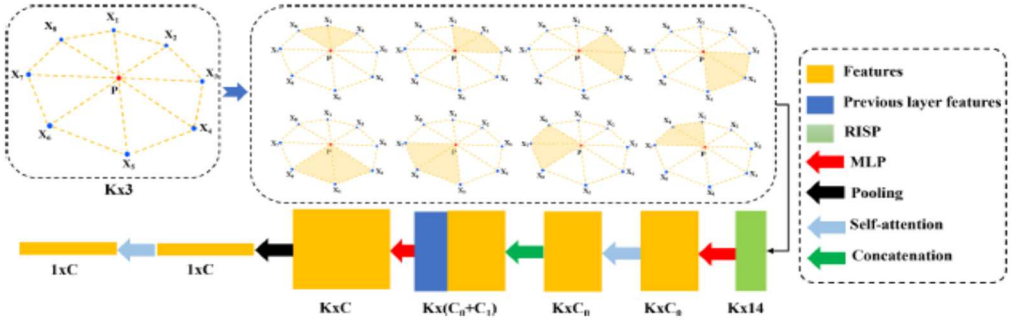


Fig. 2: RISurConv operator. For a local point set with p as the reference (red), K nearest neighbors are labelled as blue. Then, we compute the Rotation Invariant Surface Properties at each neighbor by constructing local dual triangle surfaces (Section 3), which is embedded to a high-dimensional space by a shared multi-layer perceptron (MLP) followed by a self-attention layer to produce refined features. Concatenated with previous layer features (if any), the features of these local points are further passed to MLPs, which are then summarized by maxpooling. To further refine the features, another self-attention layer follows.

Algorithm 1 RISurConv operator.

Input: Reference point p, point set Ω, point features $\mathbf{f}_{prev}$ from previous layer (if any)
Output: Convoluted features $\mathbf{f}$

$\mathbf{f} \leftarrow \{\text{RISP}(x_i) : \forall x_i \in \Omega\}$ * Construct Rotation Invariant Surface Properties (Section 3)
$\mathbf{f} \leftarrow \text{MLP}(\mathbf{f})$; * Embed each feature to a high-dimensional feature space
$\mathbf{f} \leftarrow \text{SA}(\mathbf{f})$; * Refine features via self-attention layer
$\mathbf{f}_{in} \leftarrow [\mathbf{f}_{prev}, \mathbf{f}]$ * Concatenate the features from the local and the previous layer (if any)
$\mathbf{f}_{out} \leftarrow \text{MLP}(\mathbf{f}_{in})$ * Feature embedding
$\mathbf{f}_{out} \leftarrow \text{maxpool}(\mathbf{f}_{out})$ * Maxpool features
return $\text{SA}(\mathbf{f}_{out})$ * Self-attention and return

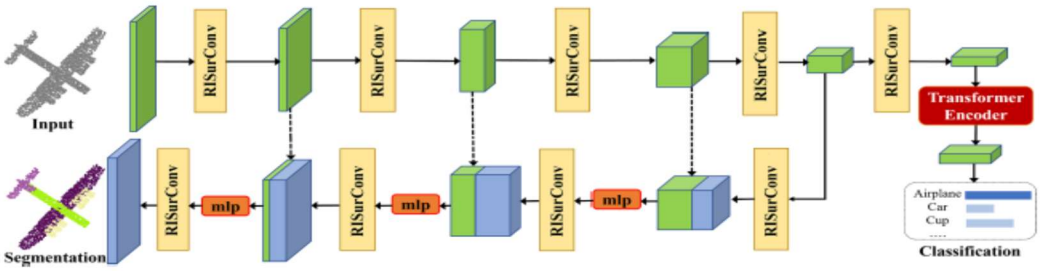


Fig. 3: Our neural network architecture comprises five RISurConv layers to extract rotation invariant features followed by a Transformer Encoder to enhance the learnt features before fully connected layers for object classification. We add a decoder with skip connections for segmentation task.

	Method	Format	Input Size	Params.
Traditional	VoxNet [21]	voxel	30^3	0.90M
	SubVolSup [24]	voxel	30^3	17.00M
	PointNet [23]	xyz	1024×3	3.50M
	PointCNN [17]	xyz	1024×3	0.60M
	PointNet++ [25]	xyz + nor	1024×6	1.40M
	DGCNN [33]	xyz	1024×3	1.84M
	RS-CNN [20]	xyz	1024×3	1.41M
	Pt Transformer [43]	xyz	1024×3	-
	Pt Transformer v2 [34]	xyz	1024×3	-
Rotation-invariant	Spherical CNN [7]	voxel	2×64^2	0.50M
	RIConv [41]	xyz	1024×3	0.70M
	SPHNet [22]	xyz	1024×3	2.90M
	SFCNN [27]	xyz	1024×3	-
	ClusterNet [3]	xyz	1024×3	1.40M
	GCAConv [40]	xyz	1024×3	0.41M
	RIF [16]	xyz	1024×3	-
	RI-GCN [15]	xyz + nor	1024×6	4.38M
	RIConv++ [42]	xyz	1024×3	0.40M
	RIConv++ [42]	xyz + nor	1024×6	0.40M
	Ours (w/o normal)	xyz	1024×3	14.0M
	Ours (w/ normal)	xyz + nor	1024×6	14.0M

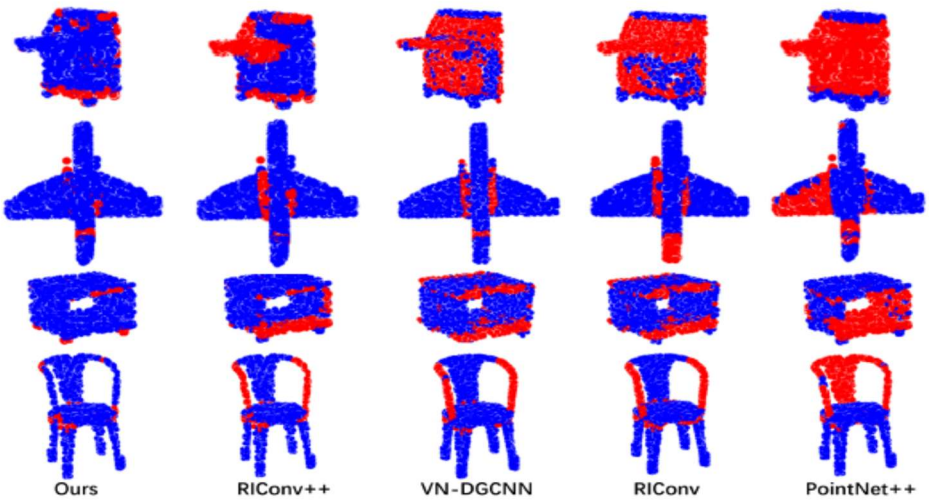
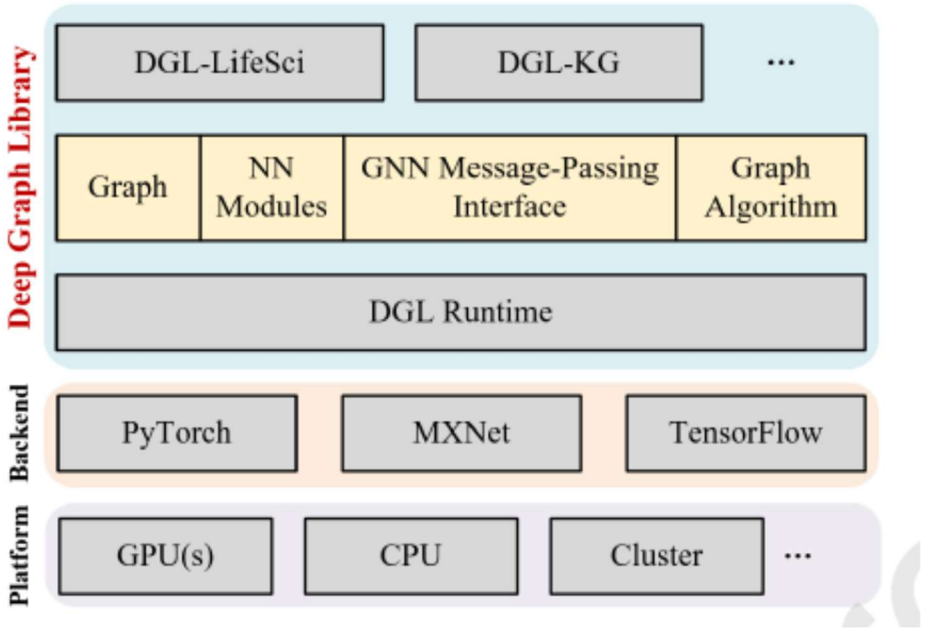


Fig. 4: Qualitative comparisons (Red indicates wrong).

Table 6: Ablation study on the Self-Attention module.

Model	SA1	SA2	Transformer Encoder	Acc.
A	✓	✓	✓	96.0
B		✓	✓	95.6
C	✓		✓	95.2
D	✓	✓		94.3
E				92.8



DGL (deep graph library) structure

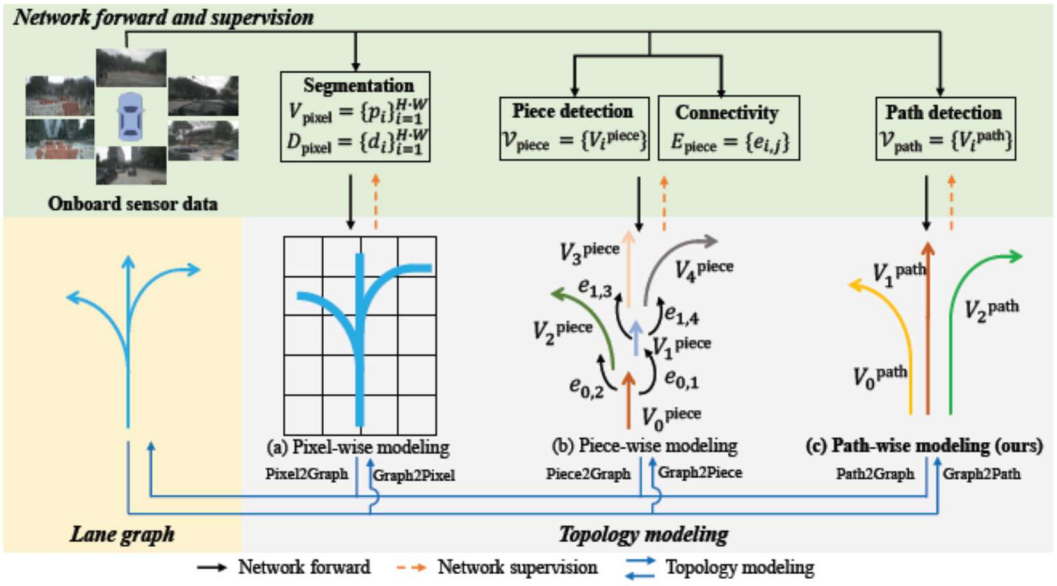


Fig. 1: Modeling comparison. **(a)** Pixel-wise modeling [22] utilizes a predefined Graph2Pixel algorithm to rasterize the lane graph into a segmentation map and a direction map on dense BEV pixels, and heuristic Pixel2Graph post-processing is needed to recover the lane graph from the predicted segmentation map V_{pixel} and direction map D_{pixel} (direction map is not drawn here for simplicity). **(b)** Piece-wise modeling [6] utilizes a predefined Graph2Piece algorithm to split the lane graph into a set of pieces and the connectivity matrix among pieces, and then it merges the predicted pieces $\mathcal{V}_{\text{piece}}$ to the graph with the Piece2Graph algorithm based on predicted connectivity E_{piece}. **(c)** The proposed path-wise modeling translates the lane graph into complete paths with a predefined Graph2Path algorithm to traverse the graph. We perform path detection and adopt a Path2Graph algorithm to recover the lane graph.

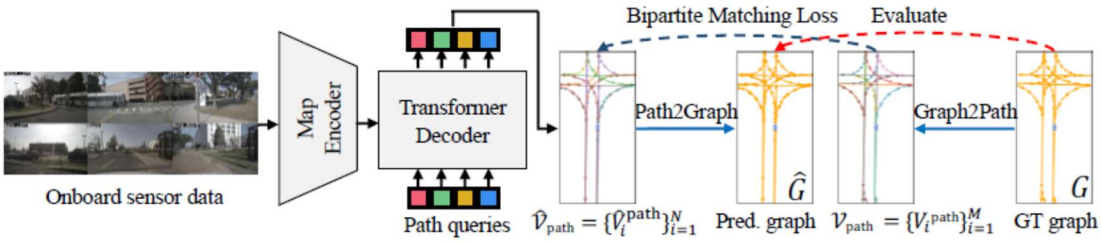


Fig. 3: Overview of LaneGAP.

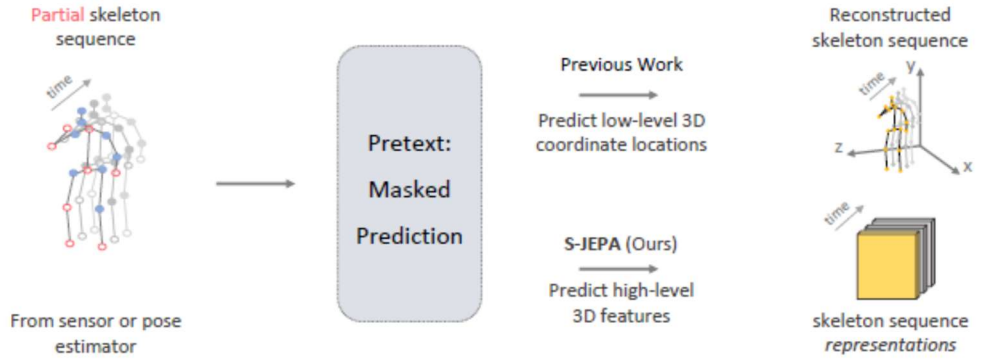


Fig. 1: Comparison between the prediction targets of previous work and S-JEPA (ours). Instead of raw 3D coordinates, S-JEPA predicts the abstract representations of 3D skeletons, embedded by a transformer encoder, effectively learning more informative high-level depth and context features for the action recognition task.

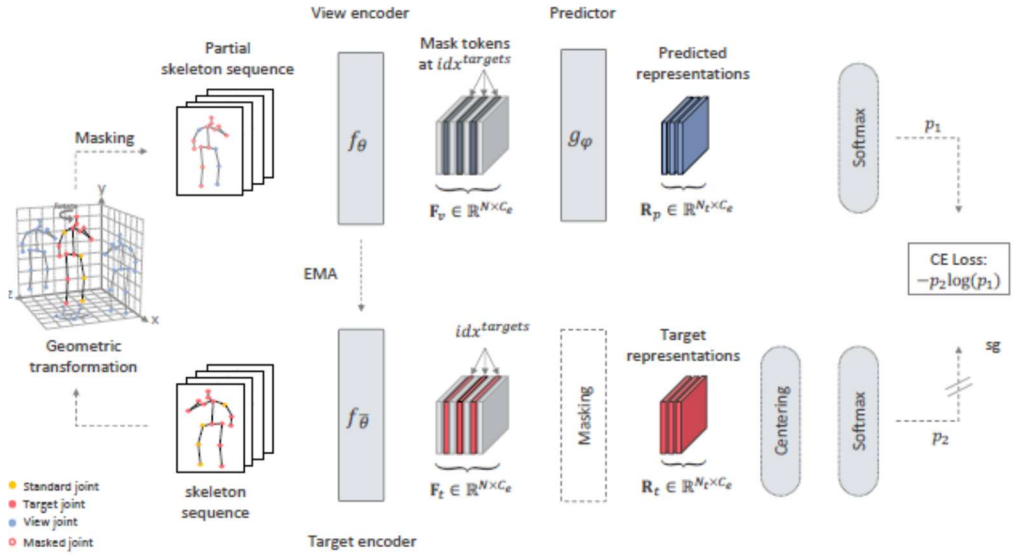
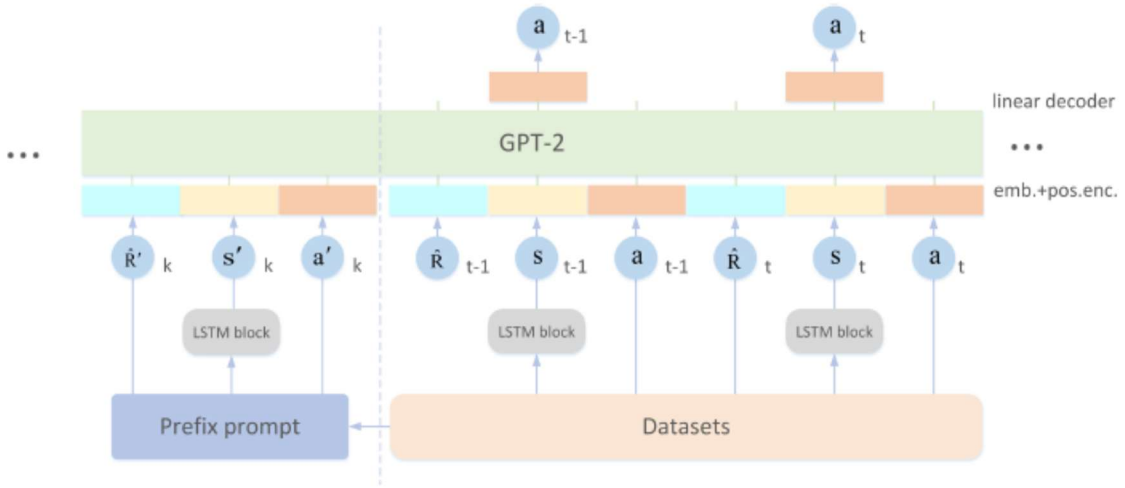


Fig. 2: Overview of S-JEPA. First, diverse skeleton views are obtained by applying geometric transformations on the 3D skeletons. The view skeletons are passed through the view encoder, after which learnable mask tokens are inserted at the locations of masked joints to get the view features $\mathbf{F}_v$. The predictor takes $\mathbf{F}_v$ as input and outputs the predicted representations $\mathbf{R}_p$ of the missing joints at the locations of the mask tokens. The target representations $\mathbf{R}_t$ are obtained by the target encoder, which takes unmasked 3D skeletons as input, and is updated through the Exponential Moving Average (EMA) of the view encoder weights after each iteration (sg denotes stop gradient). The centering and softmax operations aid in stabilizing the training loss. At fine-tuning and test times, only the target encoder weights are used.



Architecture. PLDT

Algorithm 1

Input: offline datasets D, task demonstrations P, embedding layers *embed*, transformer with causal masking *GPT*, feature processor *InnerLSTM*, learning rate α,
def $PLDT(r, s, a, t)$:
    sample trajectory τ of length K from D
    sample the highest reward segment τ' of length K' from P
    combine the two trajectory into $\tau^* = (\tau', \tau)$
    the observation o_t in the τ^* is extracted by *InnerLSTM*, get $o_t^* = InnerLSTM(o_t)$
    $\tau_{input} = stack[embed(o_t^*, a_t, r_t) + E_{pos}]$
    hidden_state = $GPT(\tau_{input})$
    hidden_action = *unstack*(hidden_state).action
    return *pred*(hidden_action)
for (r, s, a, t) in dataloader **do**
  Prediction action : $a_{pred} = PLDT(r, s, a, t)$
  Loss function : $loss = \sum(a - a_{pred})^2$
  $\theta = \theta - \alpha \nabla_\theta loss$
end for

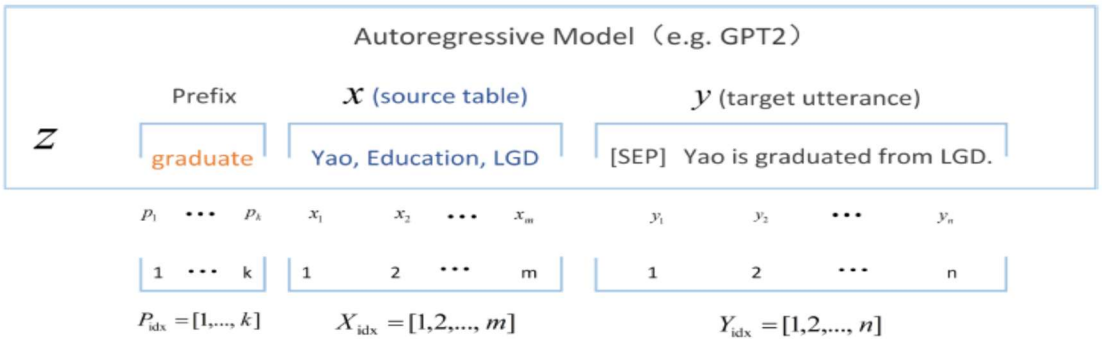


Figure 2 **Prefix tuning architecture.**

Full-size DOI: 10.7717/peerjcs.2490/fig-2

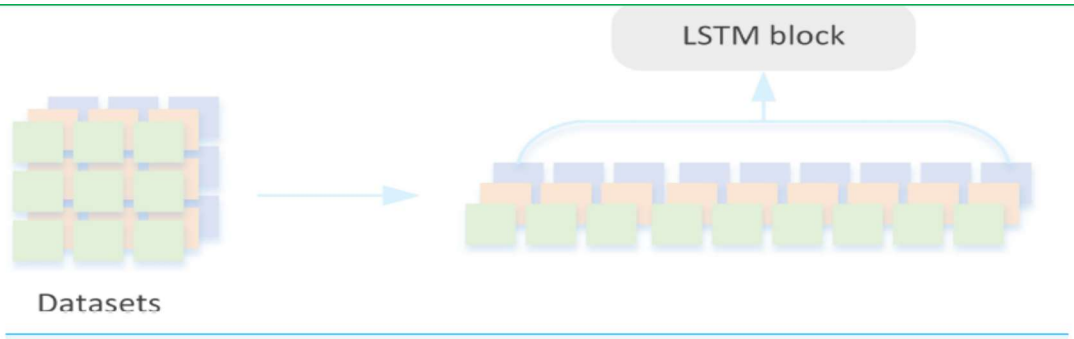


Figure 3 **LSTM block.**

Full-size DOI: 10.7717/peerjcs.2490/fig-3

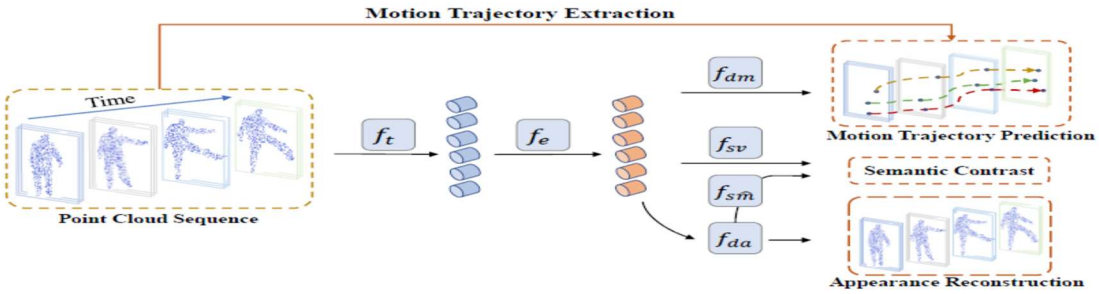


Fig. 1: The architecture of our method. We integrate three pretext tasks into the same masked autoencoder framework, i.e., motion trajectory prediction, semantic contrast, and appearance reconstruction. Note that we employ CorrNet3D to extract the motion trajectories of the point cloud sequences. f_t represents the obtaining, masking and embedding operations of the point tubes.

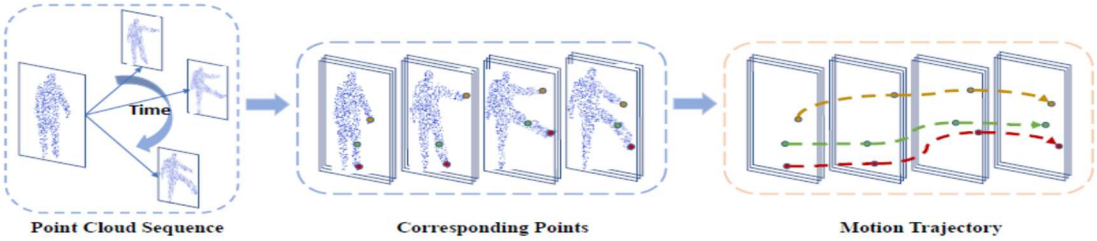
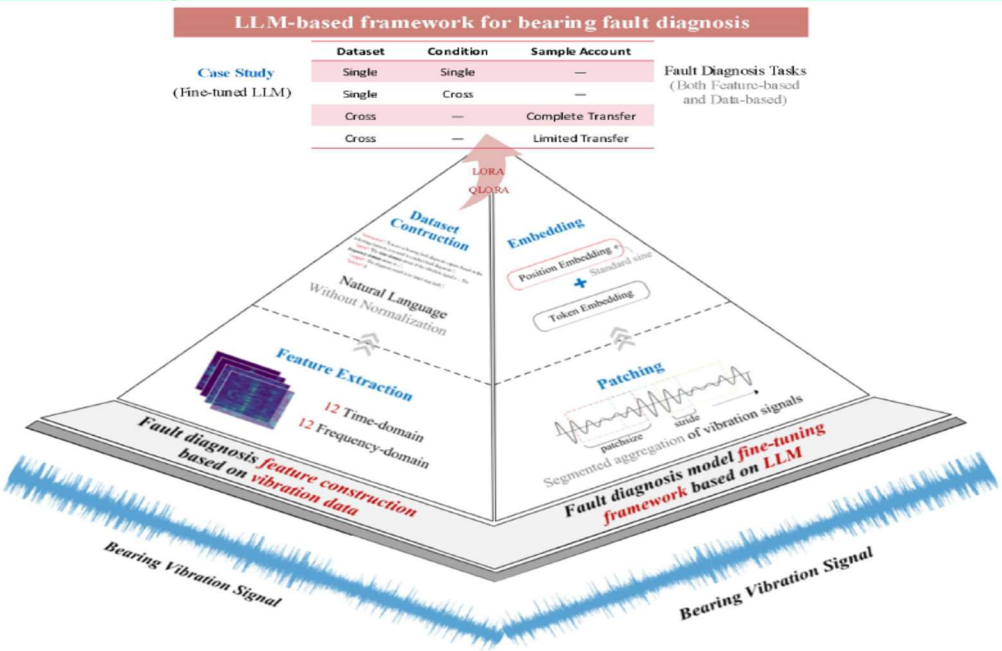


Fig. 2: Example of motion trajectory extraction of the point cloud sequence.

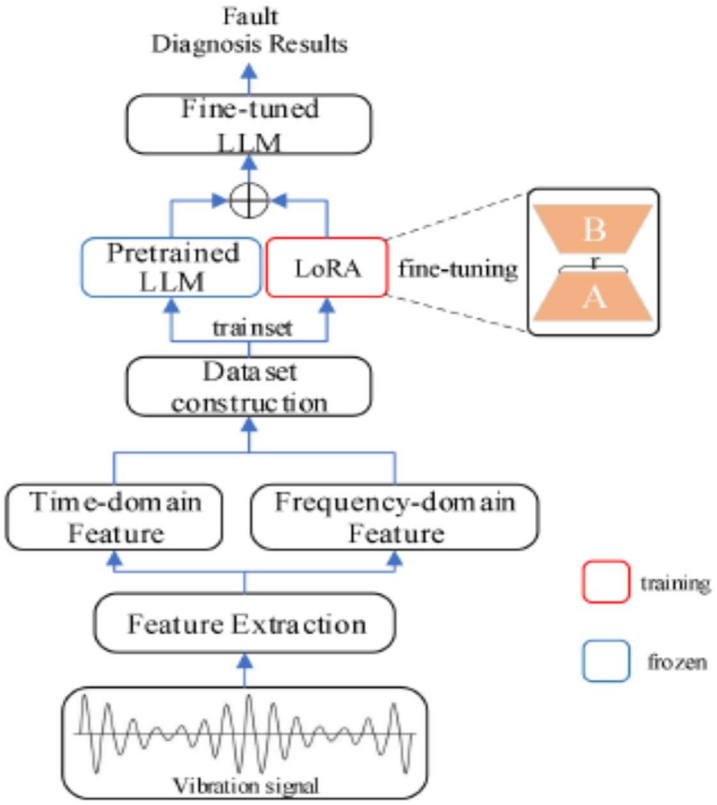
Table 1: Performance comparison of action recognition with different methods on MSRAction-3D dataset.

	Algorithm	Accuracy (%)
Supervised Learning	MeteorNet [20]	88.50
	PSTNet [9]	91.20
	PSTNet++ [10]	92.68
	Kinet [45]	93.27
	PPTr [39]	92.33
	P4Transformer [7]	90.94
	PST-Transformer [8]	93.73
End-to-end Fine-tuning	PSTNet + PointCPSC [30]	92.68
	PSTNet + CPR [29]	93.03
	PSTNet + PointCMP [28]	93.27
	P4Transformer + MaST-Pre [27]	91.29
	PST-Transformer + MaST-Pre [27]	94.08
	P4Transformer + M2PSC (ours)	**93.03**
	PST-Transformer + M2PSC (ours)	**94.84**



```
"instruction": "You are a bearing fault diagnosis expert. Based on the following features, you need to conduct fault diagnosis:",
"input": "The time-domain mean of the vibration signal is ... The frequency-domain mean is ...",
"output": "The diagnosis result is an inner ring fault.",
"history": []
```


Figure 2 Example of textualized model input



Framework of feature-based LLM fault diagnosis

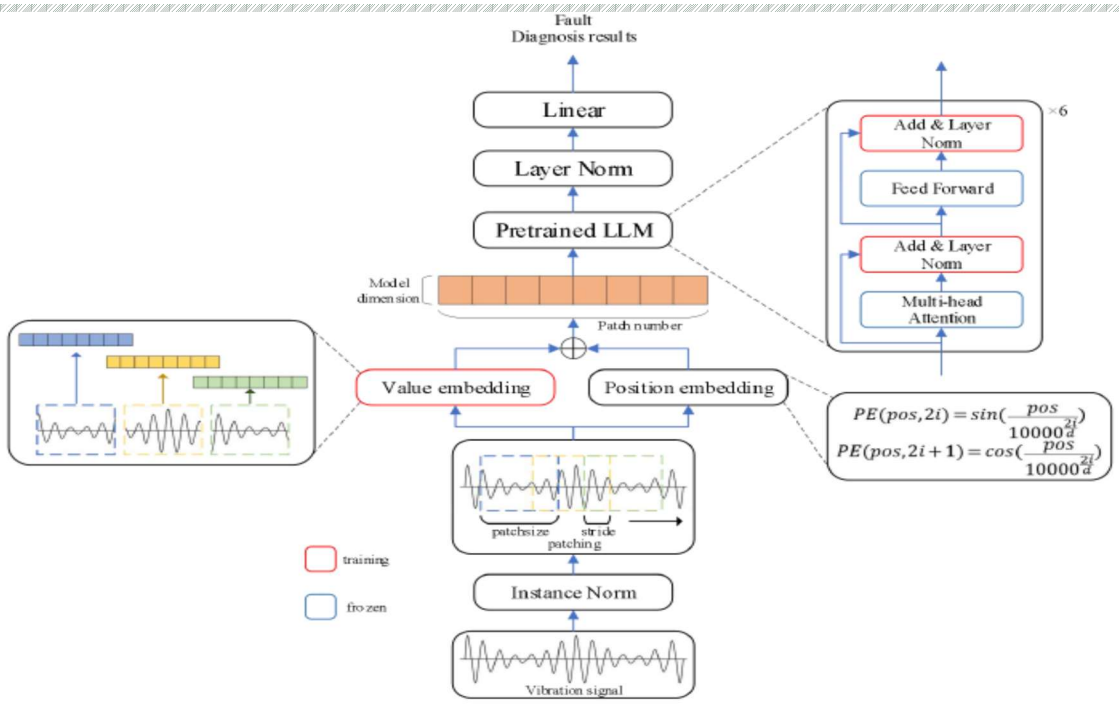
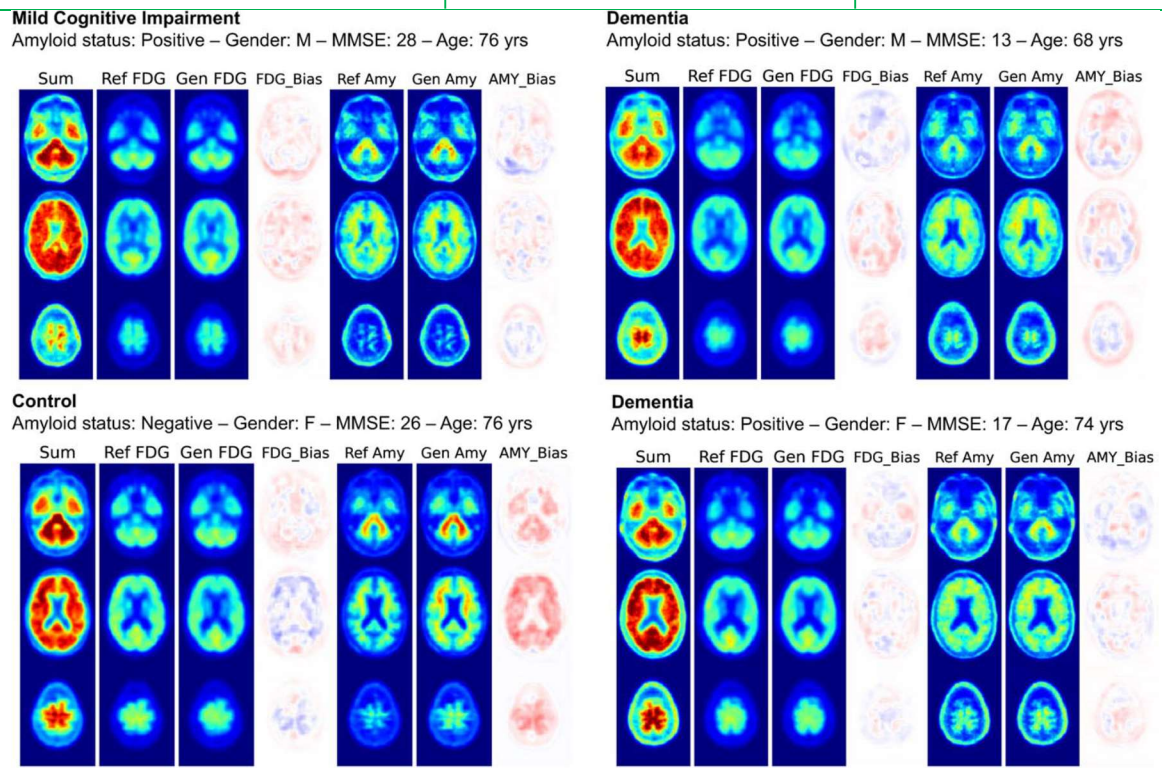


Figure 4 Framework of data-based LLM fault diagnosis

Transformer Net | **2025**

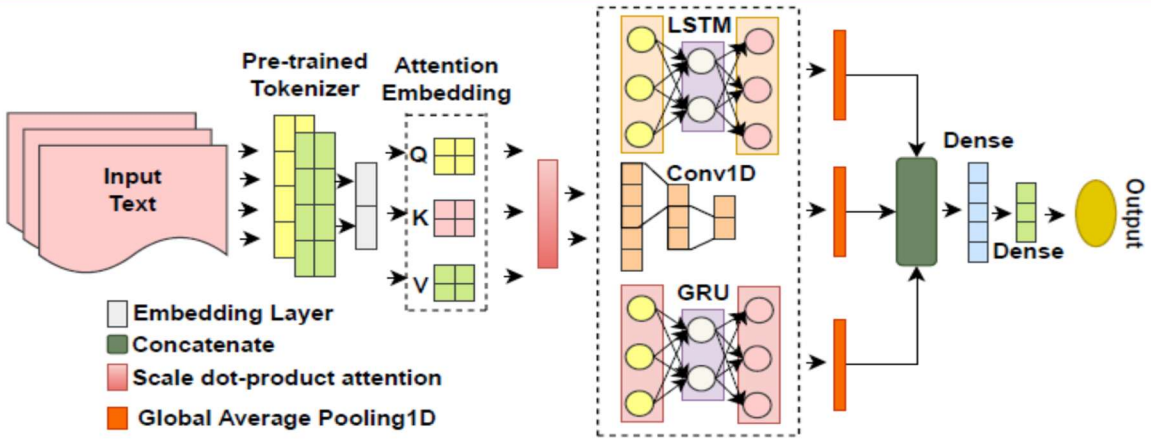


Four different subjects depicting various clinical statuses (normal, MCI, DEM) for FDG, FBP, and FMM

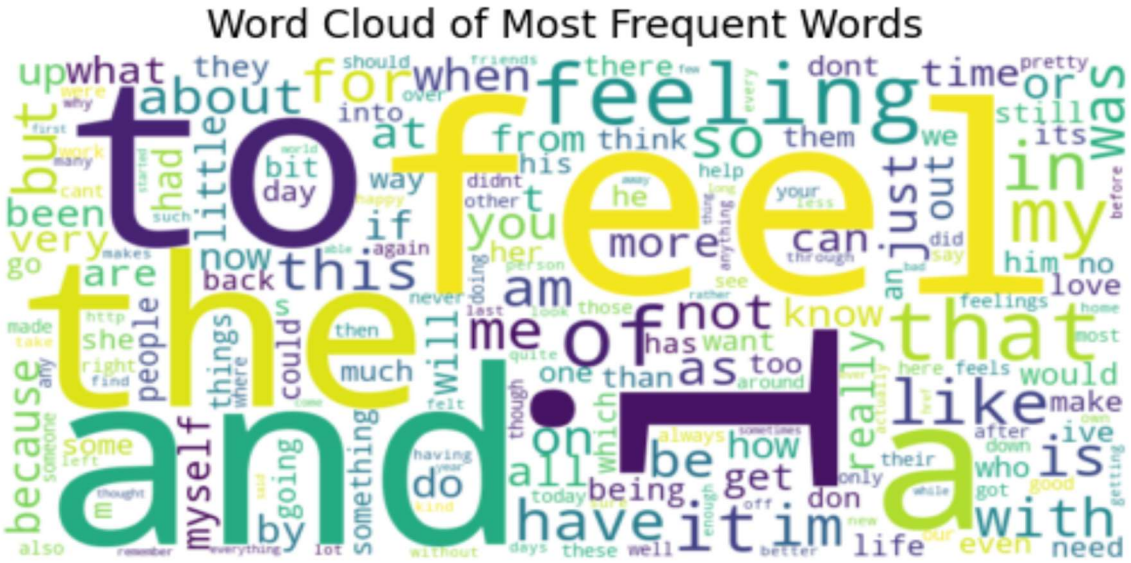
- ✓ “Sum” column displays dual tracer images (combined FDG and Amyloid [FBP and FMM]),
- ✓ “Ref FDG” represents the reference FDG
- ✓ “Gen FDG” denotes the generated FDG, and
- ✓ “FDG_Bias” signifies the difference map between reference and generated FDG
- ✓ “Ref Amy” represents reference Amyloid,
- ✓ “Gen Amy” refers to generated Amyloid, and
- ✓ “Amy_Bias” indicates difference map between reference and generated Amyloid

- The image range spans from 0 to 3 SUVR, whereas the difference map range is between −0.2 and +0.2 SUVR
- Subject-related metrics, including amyloid status, gender, MMSE, and age, are summarized atop each panel

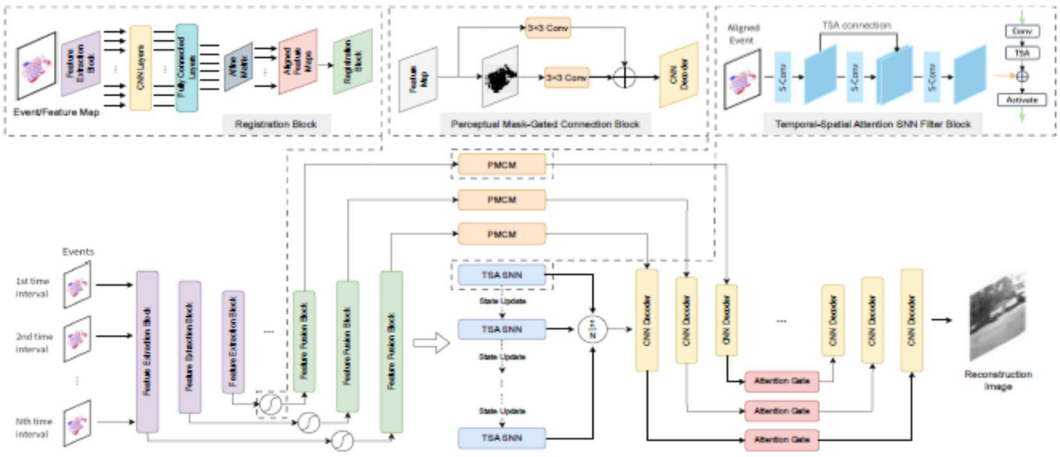
Transformer Net	**2025-79**	



EmoNet Architecture: Attention Mechanisms and Recurrent Layers for Semantic Analysis

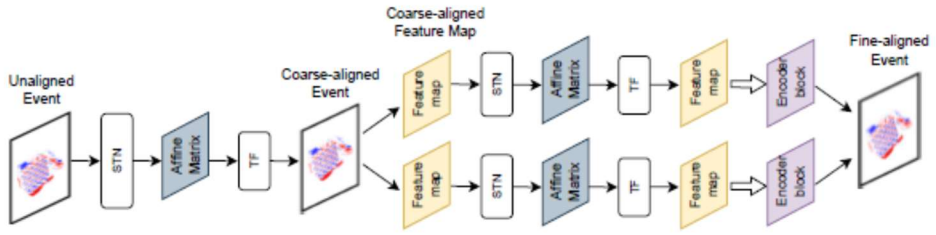


Visualization of Most Frequent Words in X Emotion Sentiment



Architecture. REDIR Network

- The model incorporates the UNet registration module with STN, and the SNN filter module, which integrates a Temporal-Spatial Attention Mechanism.
- The registered target features are then transmitted to the CNN decoder module through perceptual mask module event-based synthetic aperture imaging (E-SAI)
- REDIR: Refocus-free Event-based De-occlusion Image Reconstruction



Event Registration Module

✓ STN is utilized to predict matrices and perform affine transformations in each layer of UNet

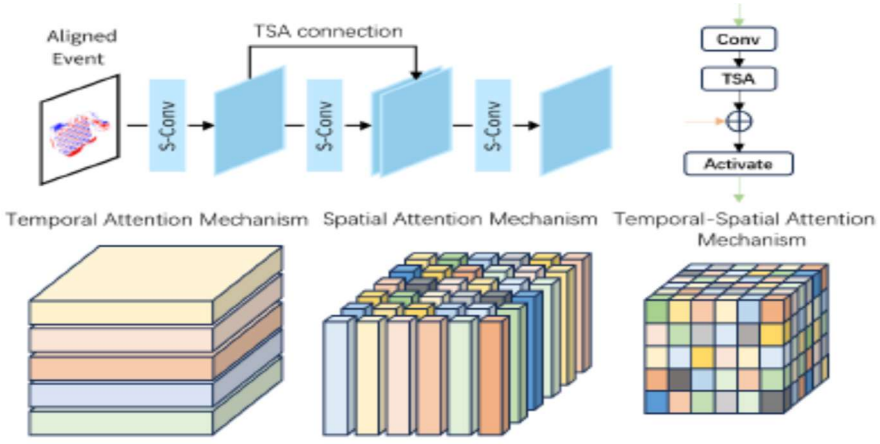


Fig. 4: TSA-SNN Occlusion Event Filtering Module

✓ SNN module incorporates a Temporal-Spatial Attention Mechanism for skip connection

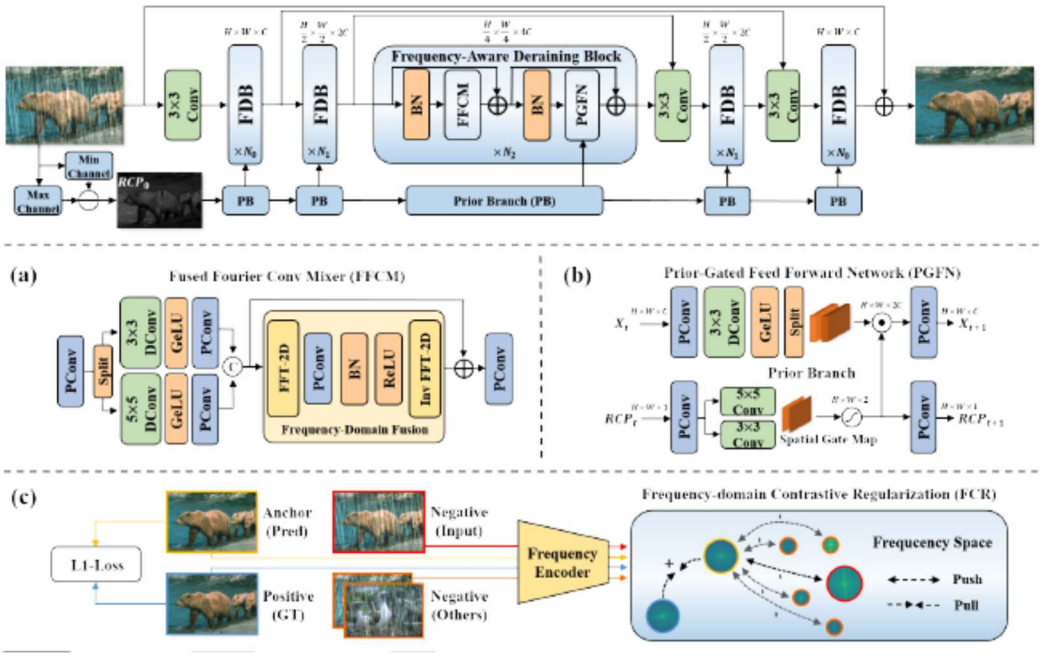


Illustration of our FADformer framework

Contains

- ✓ Fused Fourier Convolution Mixer (a, FFCM) and
- ✓ Prior-Gated Feed Forward Network (b, PGFN) with
- ✓ Frequency domain Contrastive Regularization (c, FCR)
  - For single image deraining

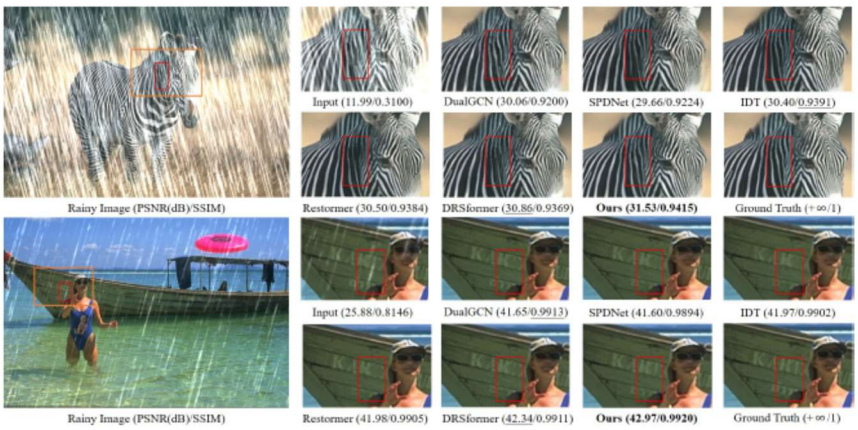
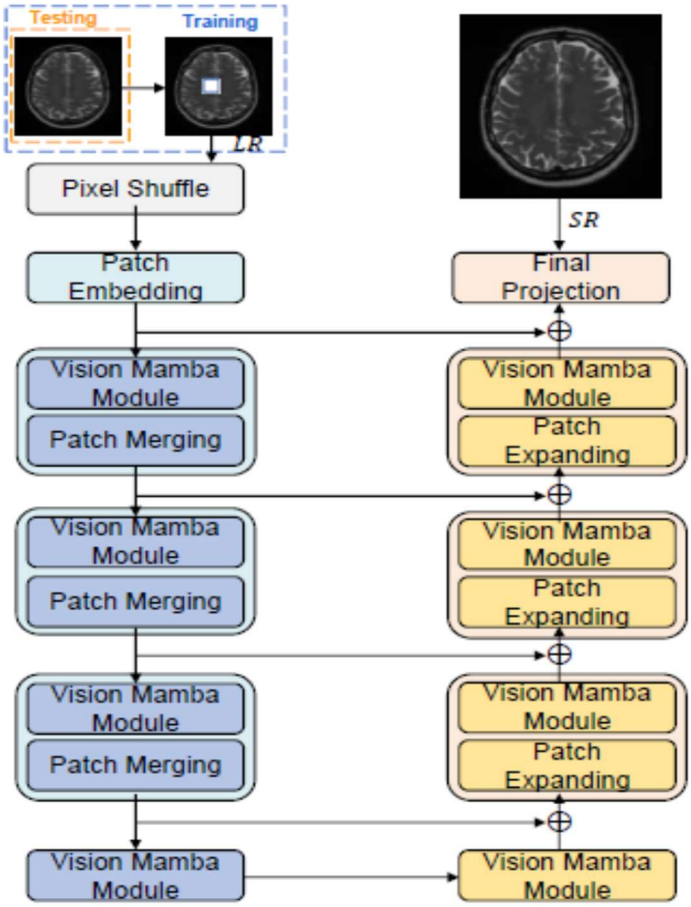
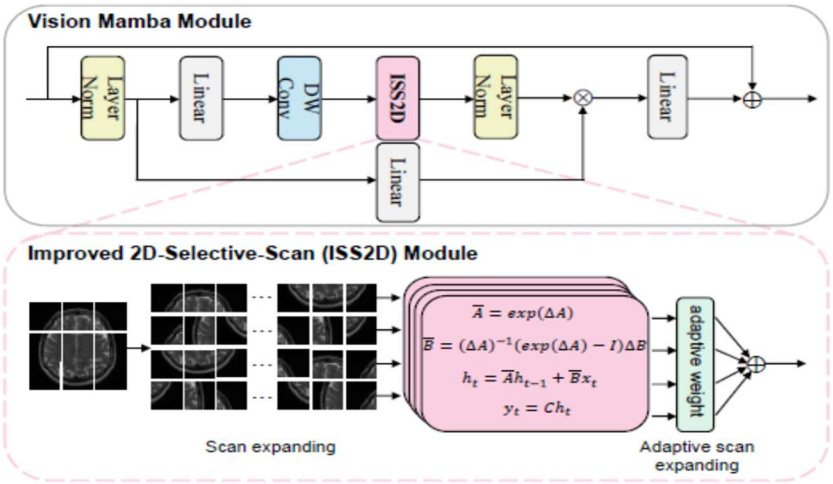


Fig. 4: Qualitative results on synthetic rainy images include the first two rows from the Rain200H and the last two from Rain200L. See supplement for more visualizations.

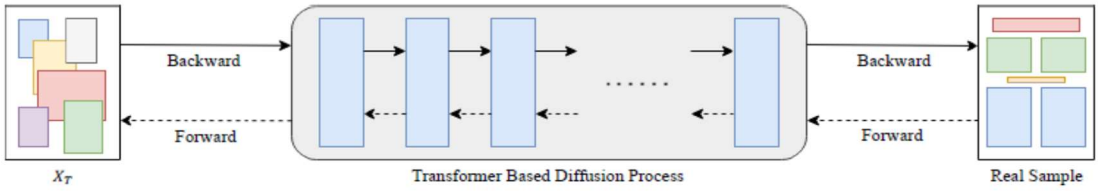


SMamba-Unet framework

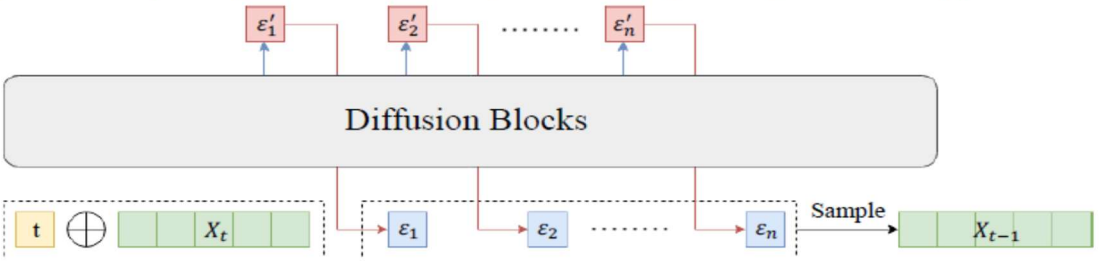
✓ Primarily comprising a patch embedding layer, a Mamba-based encoder module, a Mamba-based decoder module, and a final projection layer



Framework. vision Mamba module and improved 2D selective scan (ISS2D) module



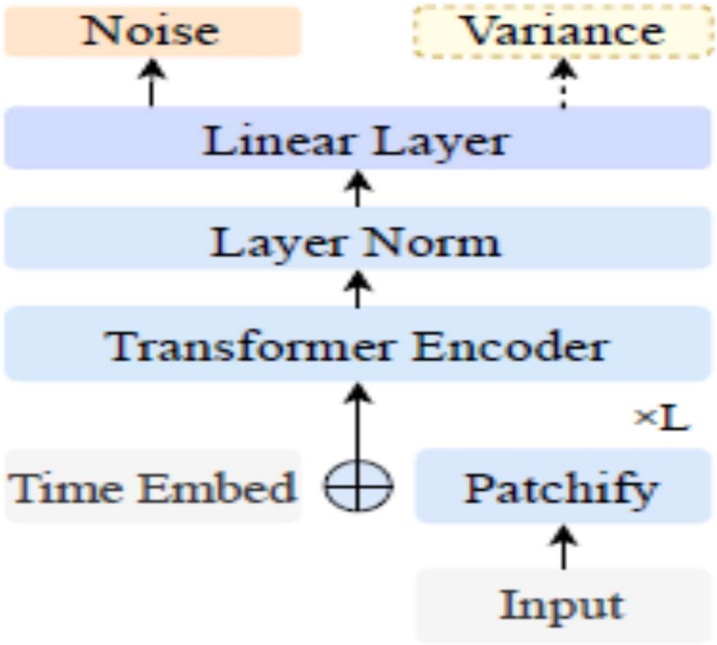
(a) **Dolfin**. The layout tensors are directly fed as input to the transformer-based diffusion block. The model processes the input and generates the desired samples without using an autoregressive decoding process.



(b) **Dolfin-AR**. The diffusion process starts with the input tensor x_t, and it passes through the transformer-based diffusion block. During each autoregressive step i, the noise ϵ_i is sampled, and both ϵ_i and other inputs are used to sample the next noise ϵ_{i+1}. Finally, the previous sample x_{t-1} is generated based on the sampled noise using DDIM (Song et al., 2022).

Dolfin model.

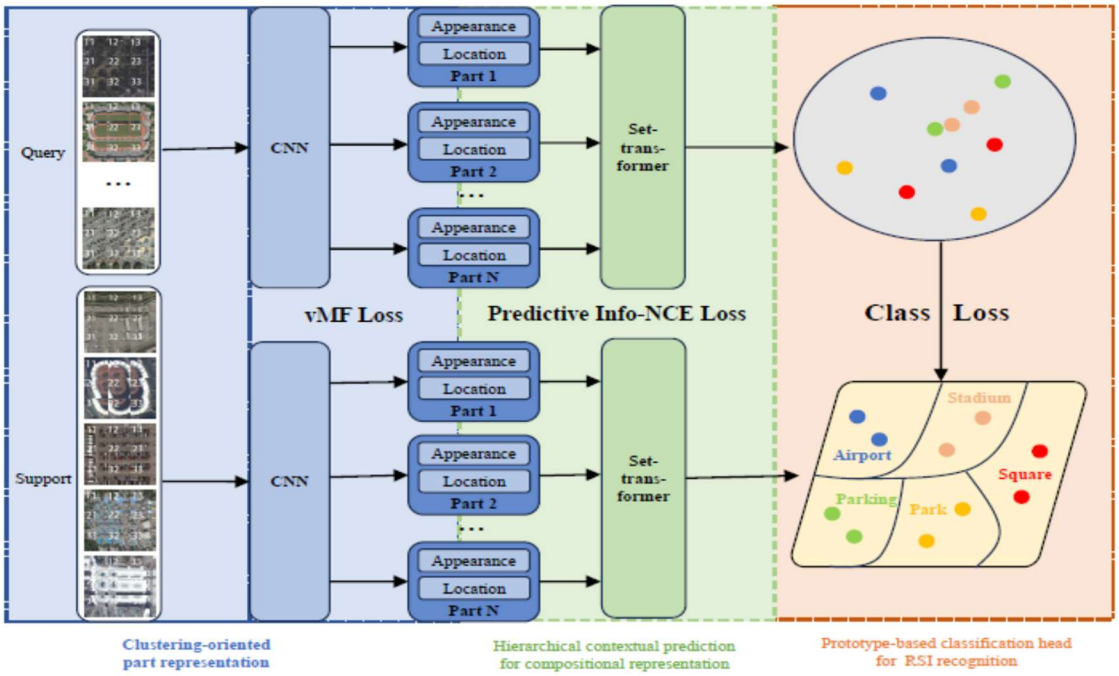
- ✓ Gaussian noise on the original input space directly applied
- DOLFIN: Diffusion Layout Transformers Without Autoencoder



Transformer structure

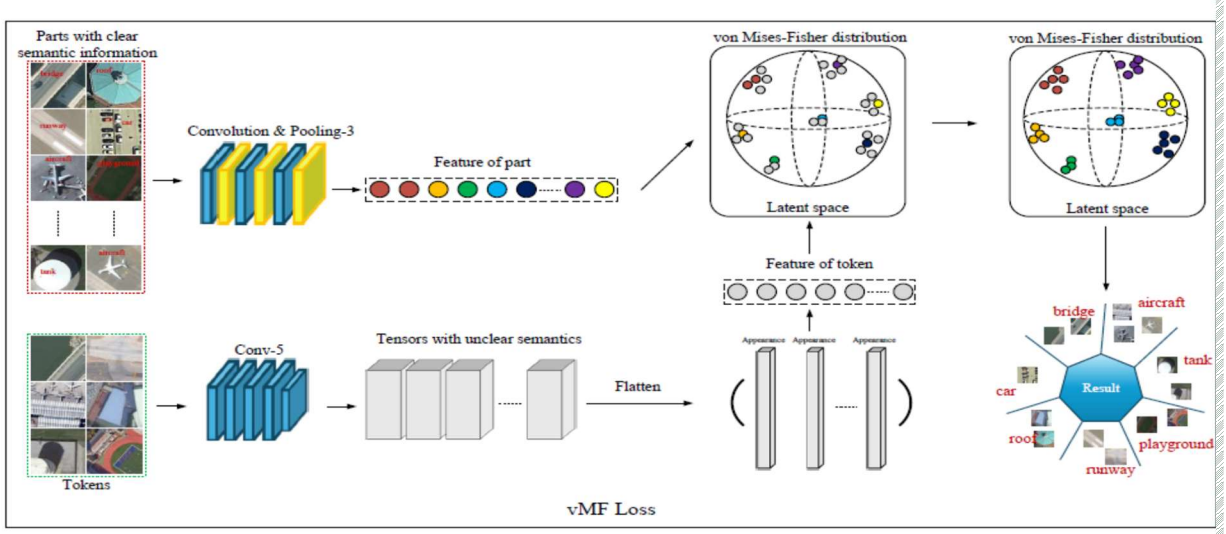
Consists of

- L transformer layers
- a layer norm and
- a linear layer.



Overview of few-shot RSIs recognition framework

- Start with dense sampling to generate discrete tokens of RSIs, to which 2D positional codes are assigned successively.
- These tokens are then clustered into distinct parts with guidance of von-Mises-Fisher (vMF) loss function, and
- The parts are combined into global remote sensing scenes through a Set-Transformer with the constraint of Predictive Info-NCE loss.
- Finally, the feature representations of global remote sensing scenes are
- classified with a prototype-based classification head.



Clustering-oriented part representation

✓ Von-Mises-Fisher (vMF) mixture model is employed to represent a multi-part distribution, where each component corresponds to a recurring part

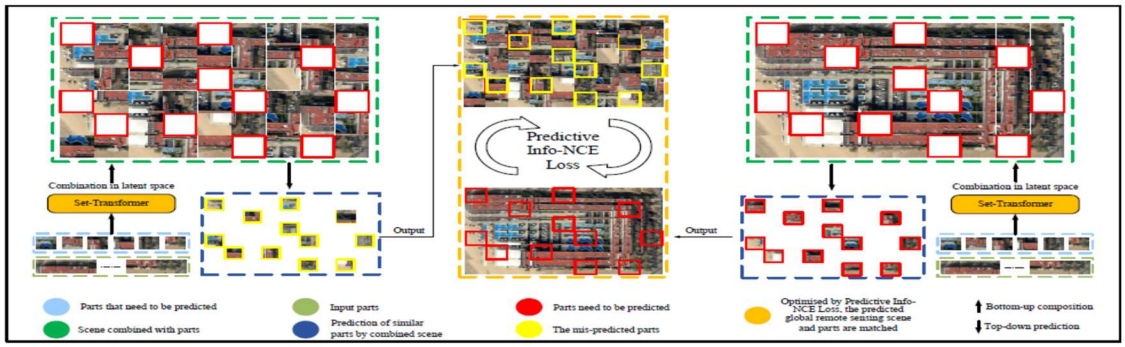
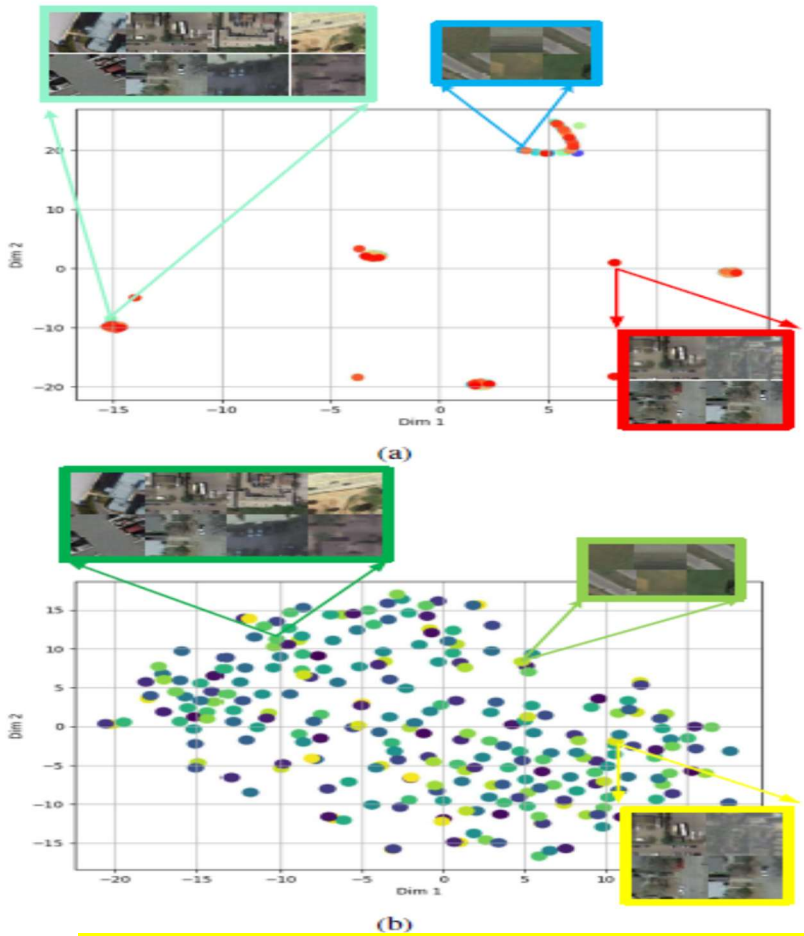


Fig. 4. Hierarchical contextual prediction for compositional representation. It consists of bottom-up composition and top-down prediction. The optimization process is constrained by the Predictive Info-NCE Loss function.



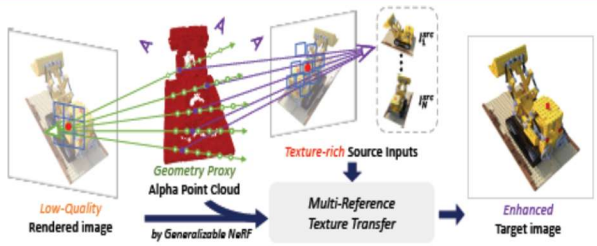
Distribution of cluster centers μ that used for AID dataset

- Each point represents a cluster centroid (i.e., part).
- Figure (a), each centroid is clustered closely, resulting in posterior representations of tokens that cannot be distinguished.
- Figure (b) depicts the scenario where each cluster centroid is separated. Here, posterior representations of the tokens can be treated as distinct parts

TABLE II
OVERALL ACCURACY(%) ON AID DATASET

Method	5-way 1-shot	5-way 5-shot
MAML [5]	43.20 ± 0.77%	60.37 ± 0.75%
MetaSGD [27]	45.01 ± 0.98%	62.58 ± 0.80%
LLSR [39]	45.18	61.76
MatchingNet [40]	33.87	50.40
RS-MetaNet [28]	58.51 ± 0.84%	73.76 ± 0.69%
DLA-MatchNet [41]	61.99 ± 0.94%	75.03 ± 0.67%
our method	**65.48 ± 0.68%**	**79.91 ± 0.41%**

Transformer Net	**2025-85**	



Geometry-driven Multi-reference Texture transfer (GMT) model

✓ GMT: Enhancing Generalizable Neural Rendering via Geometry-Driven Multi-Reference Texture Transfer

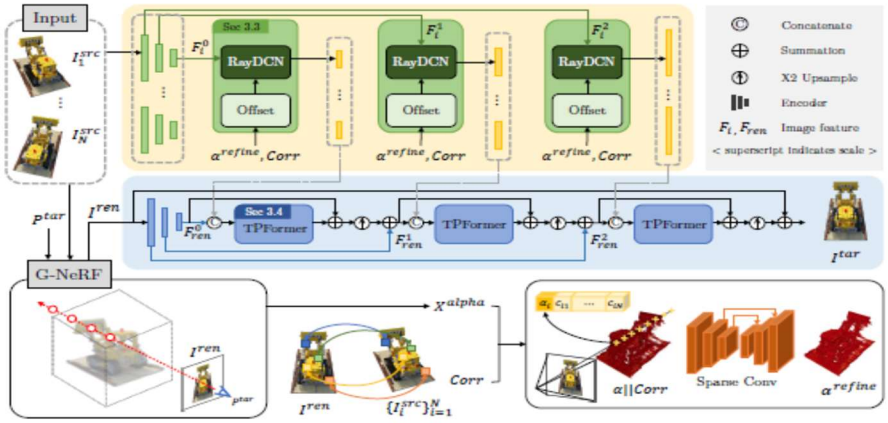
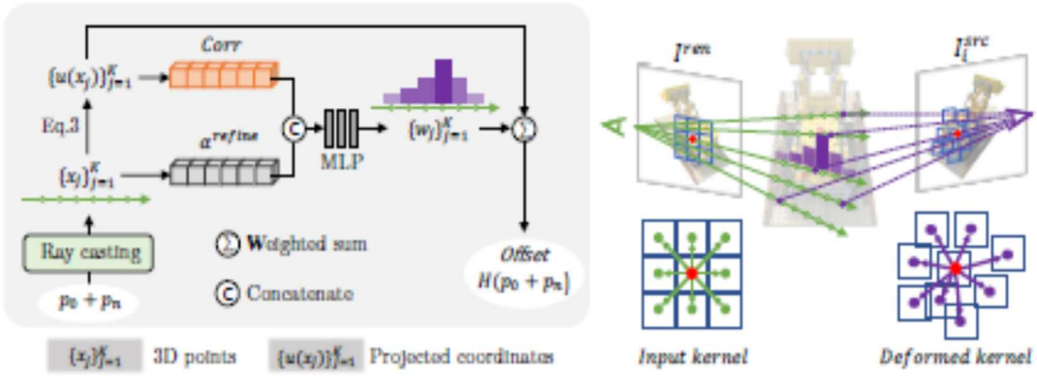
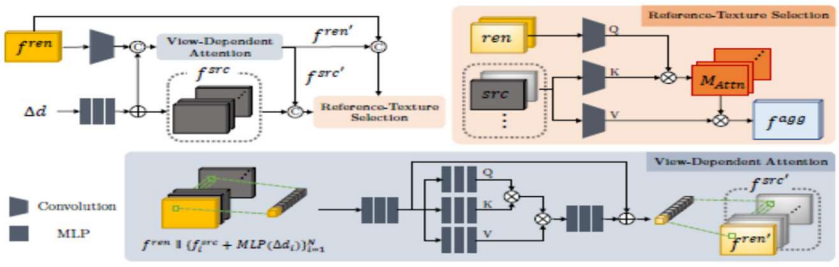


Fig. 2: Overall framework of the Geometry-driven Multi-reference Texture transfer (GMT) model. When generalizable NeRFs (G-NeRF) renders novel view image I^{ren} with N source images $\{I_i^{src}\}_{i=1}^N$ and a target camera pose P^{tar}, the process inherently generates alpha point cloud X^{alpha} for volume rendering process. Using α^{refine} extracted from α and correlation values $Corr$, RayDCN enables feature alignment considering scene geometry. Subsequently, TPFormer conducts multi-reference feature aggregation and the model generates final output I^{tar}.



Ray-imposed Deformable Convolution (RayDCN)

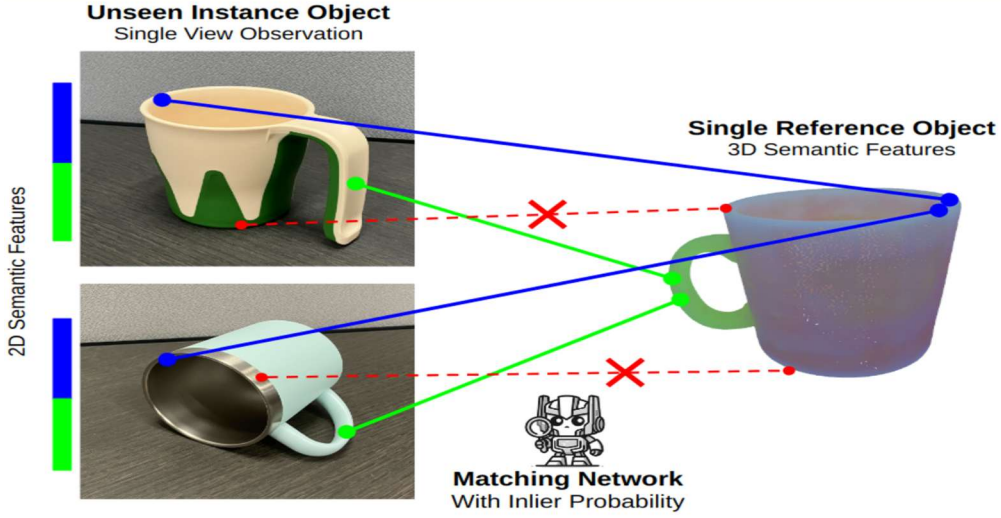
- ✓ It has a deformed kernel shape considering scene geometry and aggregates the source features of multiple rays



Texture-Preserving Transformer (TPFormer)

- ✓ TPFormer aggregates features from multiple source views while preserving textures from the source image

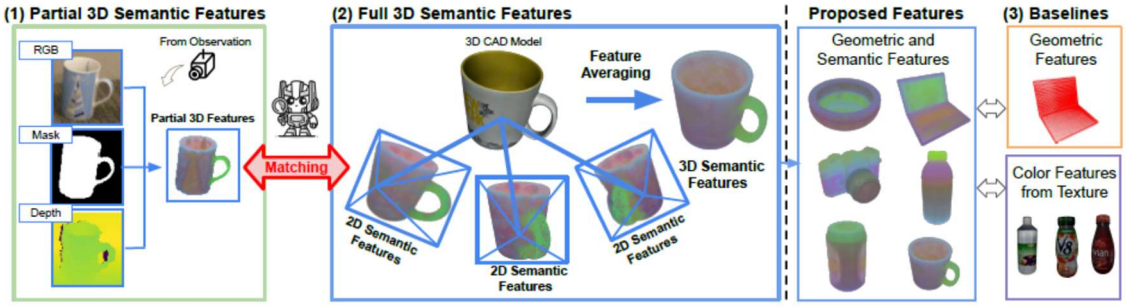
Transformer Net	2025-86	

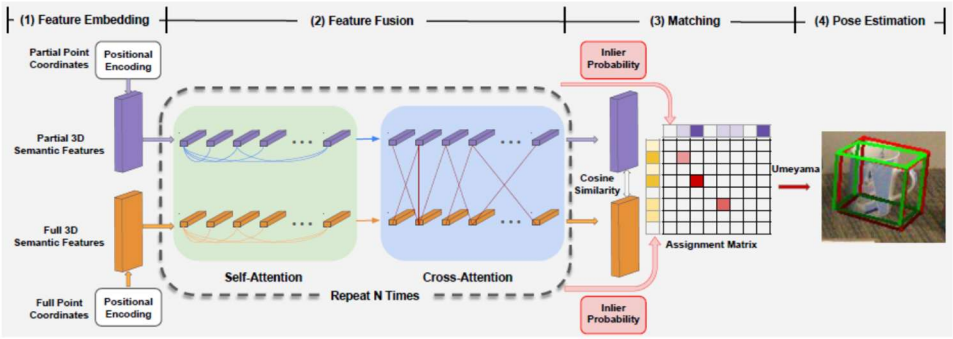


Category-level pose estimation using 3D semantic features from a pretrained foundation model.

- For a single reference object per category, 2D semantic features are projected into 3D space
- Then train a transformer matching network which is used
  - To estimate the pose of unseen objects in the category from a partial observation.

+ This approach is robust to the visual appearance of object instances

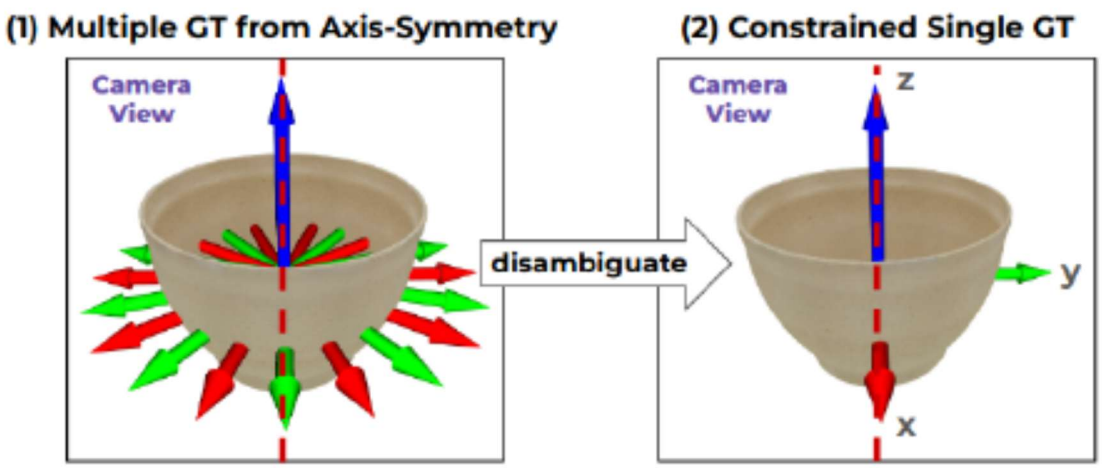




Overview of present transformer matching network

- **Objective: To match partial input and full model points with semantic features**

✓ First embed normalized point coordinates as geometric features with positional encoding and
✓ Add them with semantic features
✓ The embedded features are fused with self- and cross-attention layers for multiple iterations for global perceptions
✓ Predict the inlier probability for both partial 3D semantic features and full features, and
✓ Multiply them in the assignment matrix from cosine similarities to reduce outliers.
✓ Finally, 9D object poses of novel instances are retrieved by Umeyama algorithm [27] with RANSAC [7] from the dense correspondences



Disambiguating the symmetrical poses

- ✓ (1) Since multiple ground truth poses can exist for axis-symmetry objects
- ✓ (2) Ground Truth(GT) pose is constrained to intersect the object xz-plane with the camera origin coordinate system.

Figure 9. Visualization of predicted 3D bounding boxes on SUN RGB-D dataset [24]. Green is predicted, red is the ground truth.

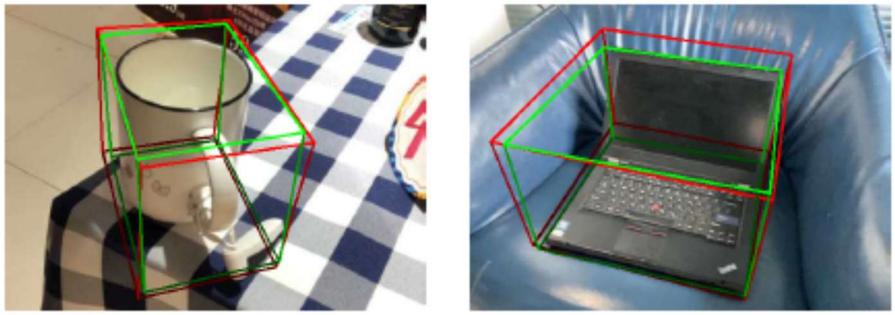


Figure 8. Visualization of predicted 3D bounding boxes on Wild6D dataset [37]. Green is predicted, red is the ground truth.

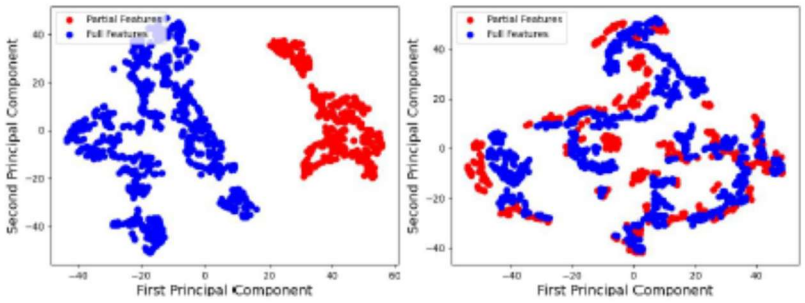
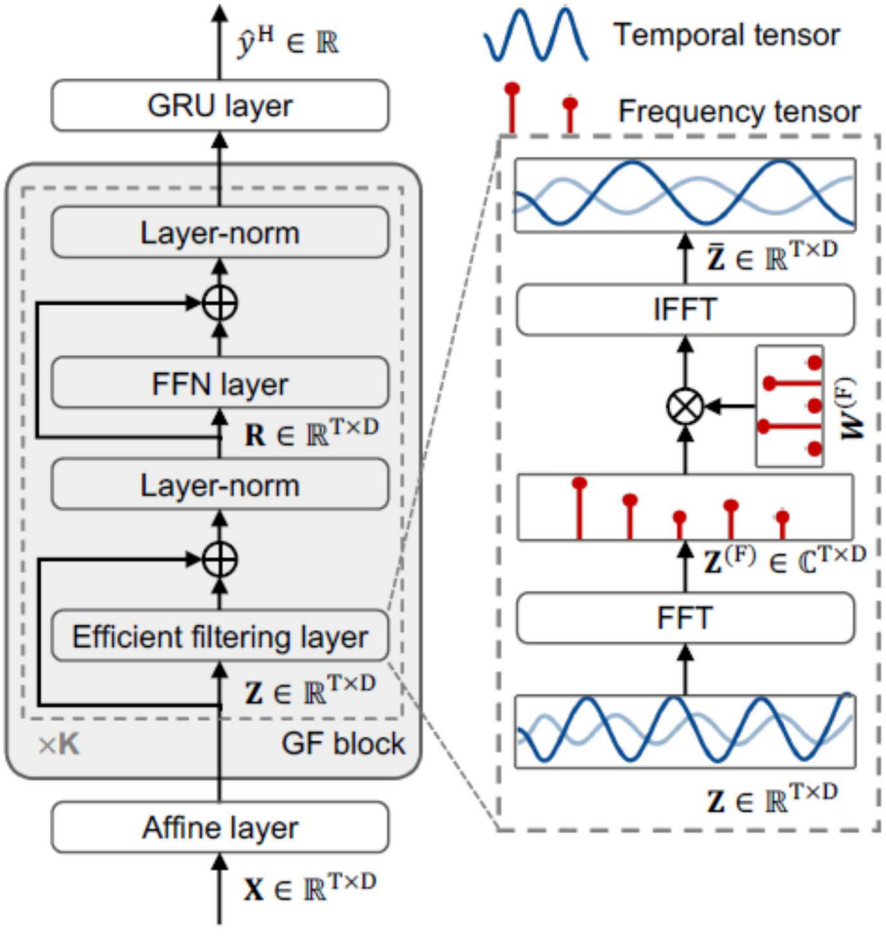


Figure 10. t-SNE visualization of 3D semantic features from partial 3D features (red) and full 3D features (blue) inside the attention region before and after feature fusion.

Transformer Net	2025-87	



Overview of the core components in DeepFilter

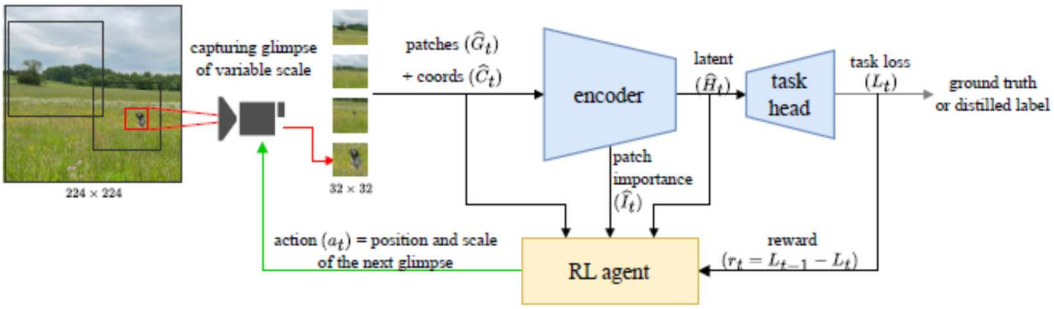
Baselines: We compare DeepFilter with three categories of baselines as follows:

- Identification methods: AR, MA and ARIMA [20];
- Statistical methods: Lasso Regression (LASSO) [23], Support Vector Regression (SVR) [41], Random Forest (RF) [21] , and eXtreme Gradient Boosting (XGB) [22];
- Deep methods: Long Short-Term Memory (LSTM) [26], Gated Recurrent Unit (GRU) [25], Transformer [28], Informer [42], AttentionMixer [7] and iTransformer [43].

✓ GRU decoder for transformer-based baselines performed to produce quality variable prediction

Methods	H=1	
	MAE	R2
Dataset: Hegang Station		
AR	$0.135_{\pm 0.000}$	$0.064_{\pm 0.000}$
MA	$0.133_{\pm 0.000}$	$0.110_{\pm 0.000}$
ARIMA	$0.135_{\pm 0.000}$	$0.063_{\pm 0.000}$
LASSO	$0.045_{\pm 0.000}$	$0.282_{\pm 0.000}$
SVR	$0.023_{\pm 0.000}$	$0.890_{\pm 0.000}$
XGBoost	$0.016_{\pm 0.000}$	$0.943_{\pm 0.000}$
LSTM	$0.021_{\pm 0.001}$	$0.846_{\pm 0.011}$
GRU	$0.023_{\pm 0.001}$	$0.832_{\pm 0.008}$
Transformer	$0.015_{\pm 0.002}$	$0.927_{\pm 0.022}$
Informer	$0.042_{\pm 0.017}$	$0.562_{\pm 0.269}$
iTransformer	$0.014_{\pm 0.001}$	$0.919_{\pm 0.026}$
AttentionMixer	$\mathbf{0.011}_{\pm 0.003}$	$\underline{0.962}_{\pm 0.012}$
DeepFilter	$\underline{0.012}_{\pm 0.001}$	$\mathbf{0.963}_{\pm 0.006}$
Dataset: Jinan Station		
AR	$0.106_{\pm 0.000}$	$0.744_{\pm 0.000}$
MA	$0.112_{\pm 0.000}$	$0.710_{\pm 0.000}$
ARIMA	$0.101_{\pm 0.000}$	$0.759_{\pm 0.000}$
LASSO	$0.029_{\pm 0.000}$	$0.742_{\pm 0.000}$
SVR	$0.061_{\pm 0.000}$	$0.640_{\pm 0.000}$
XGBoost	$0.031_{\pm 0.000}$	$0.903_{\pm 0.000}$
LSTM	$0.015_{\pm 0.000}$	$0.948_{\pm 0.001}$
GRU	$0.016_{\pm 0.001}$	$0.939_{\pm 0.005}$
Transformer	$0.016_{\pm 0.004}$	$0.955_{\pm 0.023}$
Informer	$0.018_{\pm 0.002}$	$0.917_{\pm 0.013}$
iTransformer	$0.012_{\pm 0.000}$	$0.971_{\pm 0.005}$
AttentionMixer	$\underline{0.011}_{\pm 0.004}$	$\underline{0.981}_{\pm 0.012}$
DeepFilter	$\mathbf{0.010}_{\pm 0.002}$	$\mathbf{0.986}_{\pm 0.003}$

Comparative Study On The Hegang And Jinan Datasets Over Four Forecast Horizons



Architecture: AdaGlimpse

Consists of two parts:

- A vision transformer-based Encoder with a task-specific head (see Sec. 3.2) and a
- Soft Actor-Critic RL agent (see Sec. 3.3).

✓ At each exploration step, the RL agent selects the position and scale of the next glimpse based on the information about previous patches, their coordinates, importance, and latent representations

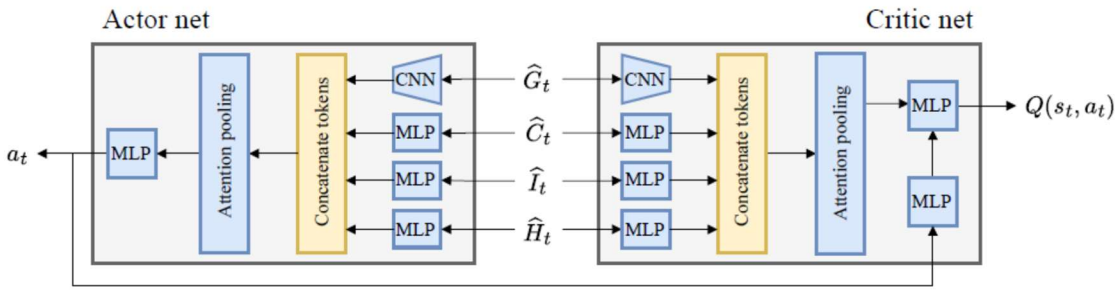


Fig. 3: RL agent: RL module of AdaGlimpse uses two networks: the actor and the critic. The actor predicts the action a_t (position and scale of the next glimpse) based on state $s_t = (\widehat{G}_t, \widehat{C}_t, \widehat{I}_t, \widehat{H}_t)$. The critic estimates the $Q(s_t, a_t)$, corresponding to the expected cumulative reward for taking this action.

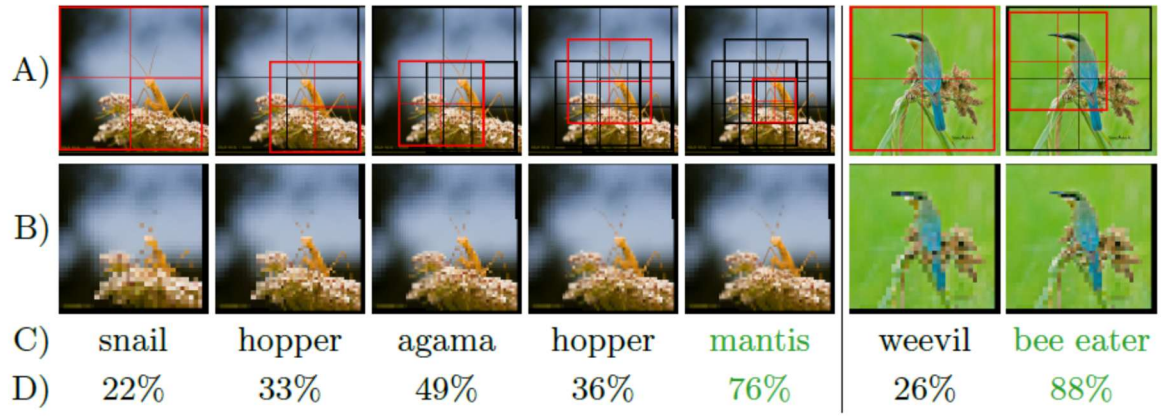


Fig. 4: Glimpse selection step-by-step: AdaGlimpse explores 224×224 images from ImageNet with 32×32 glimpses of variable scale, zooming in on objects of interest and stopping the process after reaching 75% predicted probability. The rows correspond to: A) glimpse locations, B) pixels visible to the model (interpolated from glimpses for preview), C) predicted label, D) prediction probability.

Model and training configuration

✓ **Model and training configuration details specified**

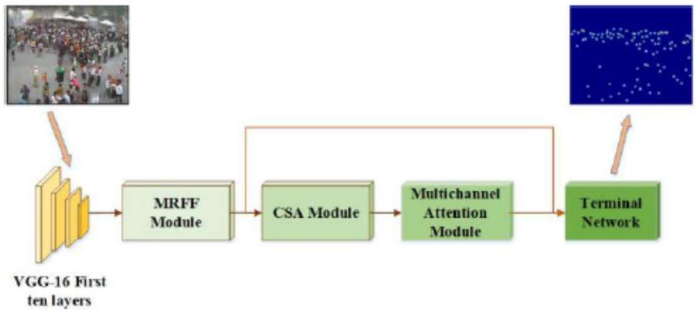
Parameter name	Value
Encoder	
ViT type	base
native patch size	16
transformer embed dim	768
transformer blocks	12
attention heads	12
mlp ratio	4
Decoder	
transformer embed dim	512
transformer blocks	8
attention heads	16
mlp ratio	4
RL agent	
hidden dim	256
action distribution	TanhNormal
sac target entropy value	−3
sac initial alpha value	1

Training	
training epochs	100
backbone pre-training epochs	600
backbone lr	10^{-5}
rl agent lr	$5 * 10^{-4}$
backbone weight decay rate	10^{-4}
rl agent weight decay rate	10^{-2}
lr scheduler type	one cycle
lr warmup epochs	10
minimum lr	10^{-8}
initial random action batches	10000
initial frozen backbone epochs	10
rl loss function	L2
rl batch size	256
backbone batch size	128
replay buffer size	10000

Transformer Net | **2025-89**

Three images from the ShanghaiTech PartA test set

- ✓ Left to right: scenario problems for the three counting tasks:
- ✓ Scale change, occlusion, and crowded places



CSFNet general framework

- Terminal Network is {C(512,512,1) - C(512,512,1) - C(512,256,1) - C(256,128,1) - C(128,64,1) - C(64,1,1)},
- where C(I,O,K) denotes a convolutional layer with an input channel I, an output channel O, and the convolutional layer with kernel size K

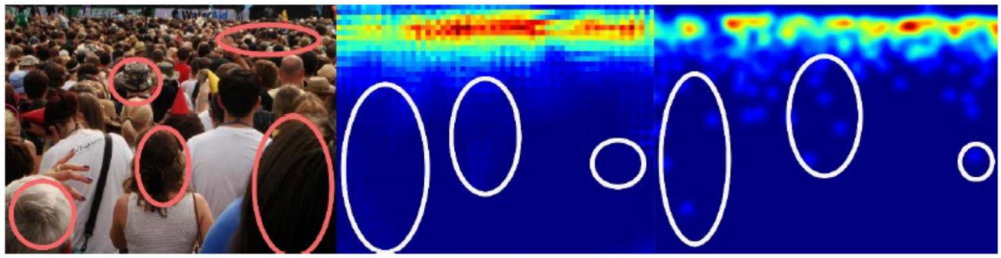
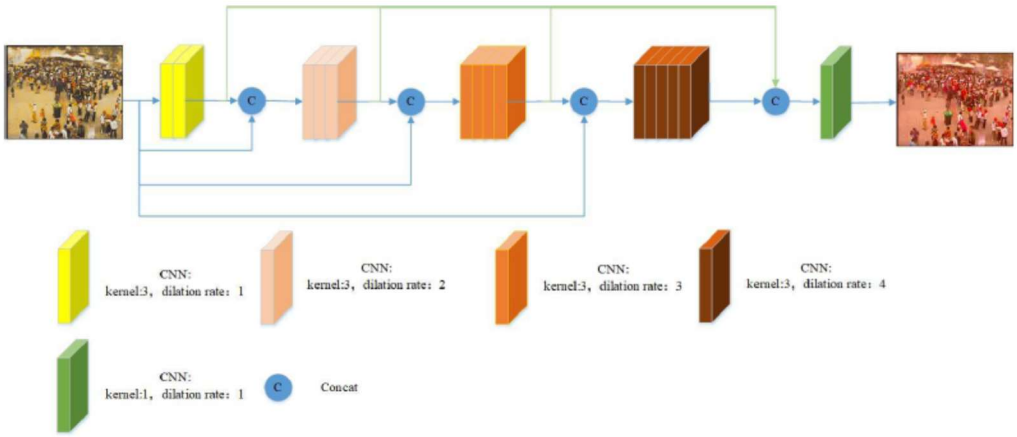
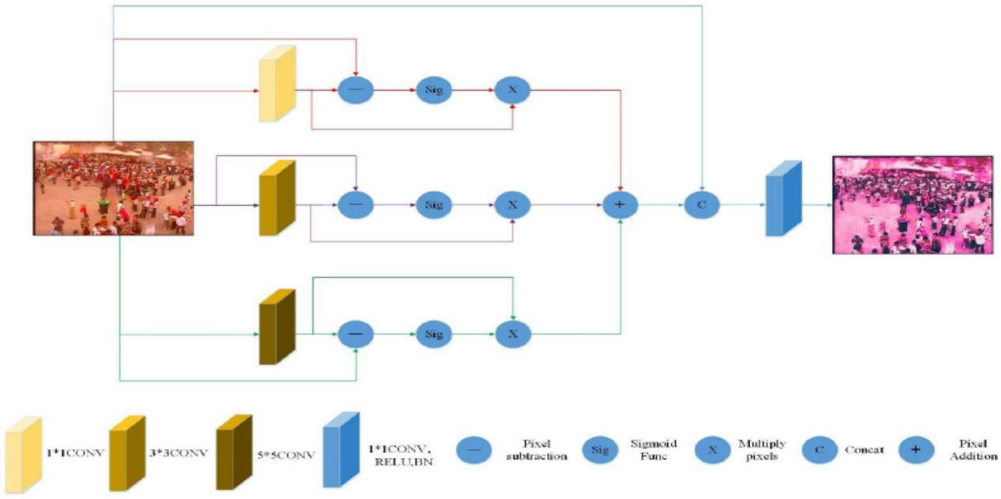


Image from the Shanghai parta dataset, complete with corresponding predicted and True Density plots.

Red circles : Variation in head size
White circles : Decrease in counting accuracy due to the difficulty in coping with the scale variation during the test



General framework:MRFF Module

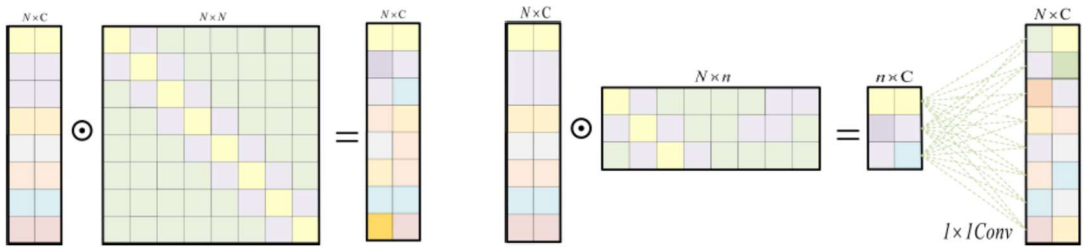


General framework CSA Module

Table 2: Comparison of CSFNet with other advanced counting methods in ShanghaiTech dataset

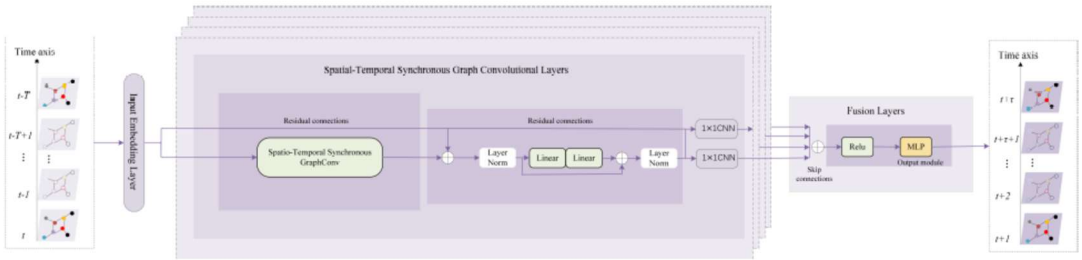
Method	Part A		Part B	
	MAE	MSE	MAE	MAE
MCNN[10]	110.2	173.2	26.4	41.3
CSRNet[14]	68.2	115.0	10.6	16.0
DSPNet[37]	68.2	107.8	8.9	14.0
C-CNN[37]	88.1	141.7	14.9	22.1
DNCL[37]	73.5	112.3	18.7	26.0
ED-CNN[40]	69.8	114.7	10.2	14.9
ICC[42]	76.9	130.1	8.4	15.2
TransCrowd-token[43]	69.0	116.5	10.6	19.7
TransCrowd-GAP[43]	66.2	105.1	9.3	16.1
AutoScale_loc[44]	**65.8**	112.1	8.6	13.9
MACC+SM[44]	67.7	113.0	9.8	12.9
MSC-FFN[32]	**65.8**	105.9	7.6	**11.8**
CCD Net[46]	70.0	118.3	-	-
DA^{2}Net[33]	74.1	128.4	7.9	13.2
T^{2}CNN[47]	85.3	137.4	18.8	29.2
CSFNet(ours)	66.1	**103.2**	**7.5**	**11.8**

Transformer Net | **2025-92**

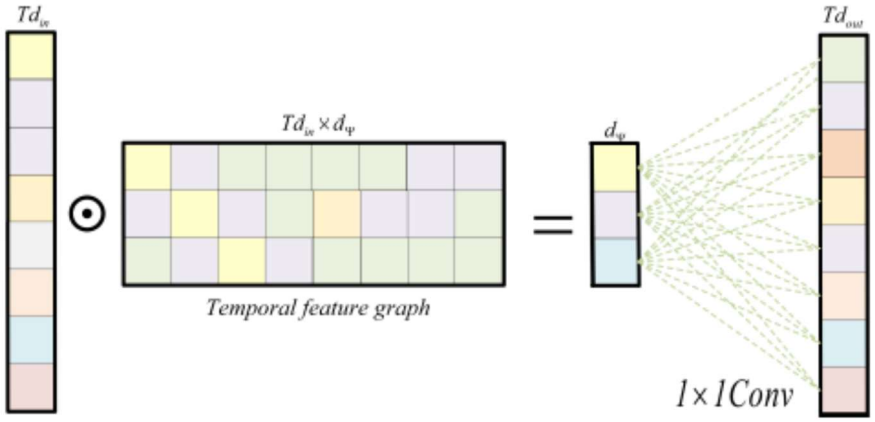


(a) The process of traditional graph computation (b) The process of Faster graph computation

Process of graph computation with different methods



Architecture. FasterSTS



Computation process of the spatio-temporal Synchronous graph convolution kernel

Transformer Net	**2025-93**	

Figure 1: Demonstrating rotation-invariant face detection in complex, real-life scenarios with inevitable in-plane face rotations captured during gatherings, sports, and artistic performances.

RP-Net: A Robust Polar Transformation Network for Rotation-Invariant Face Detection

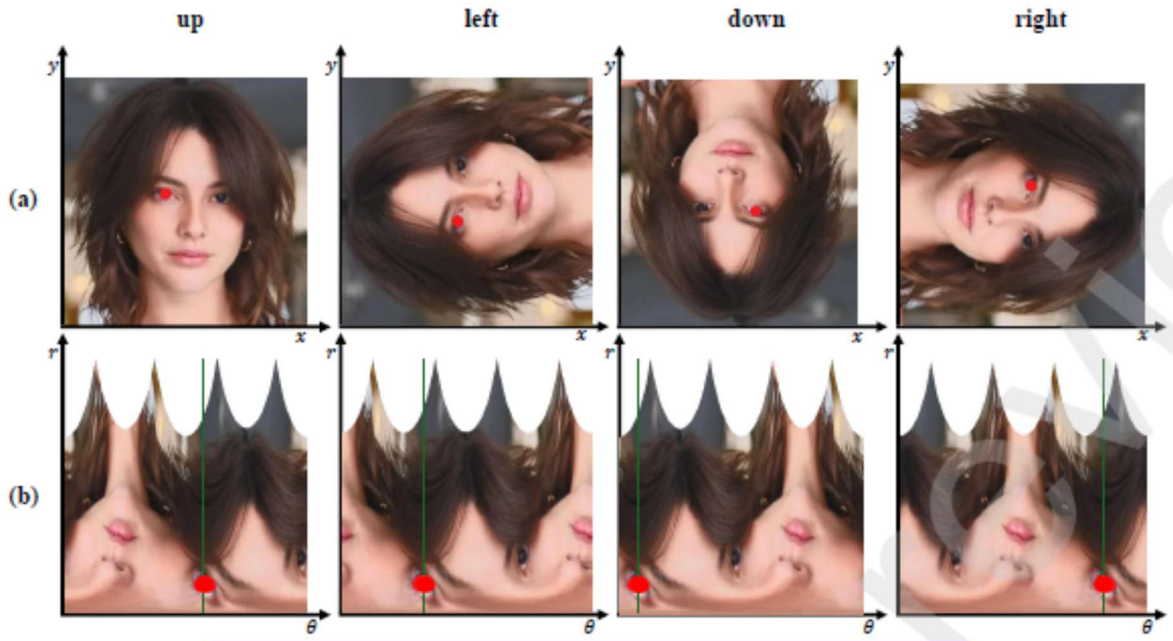
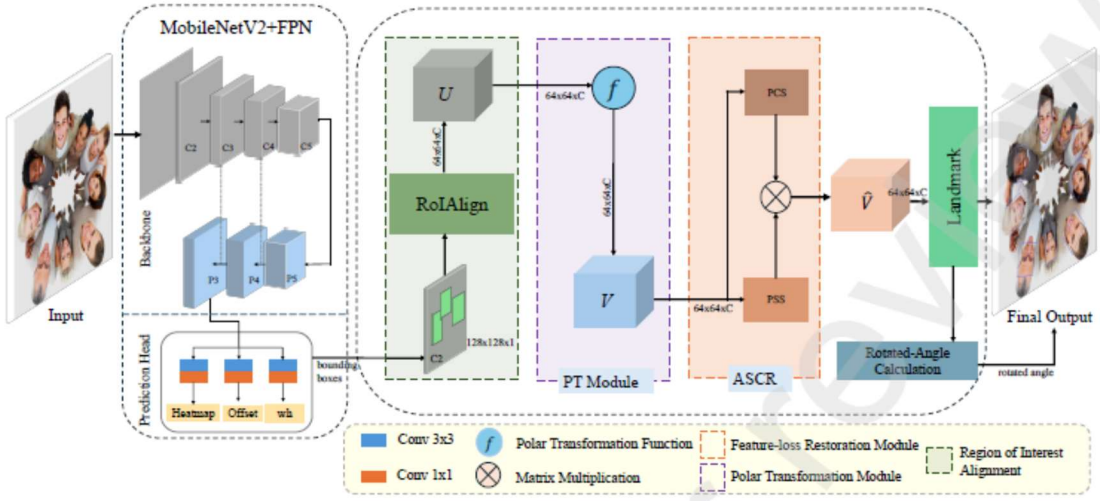


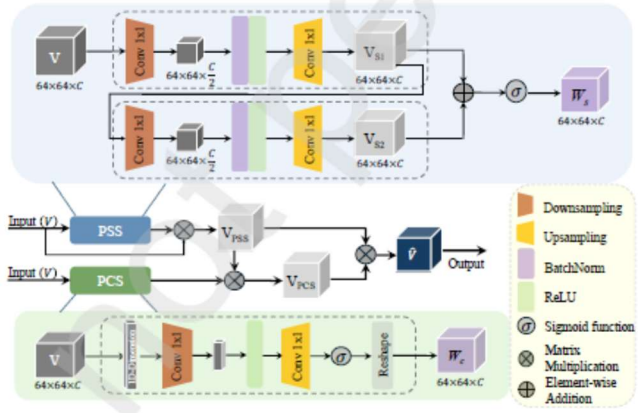
Illustration of polar transformation applied to a rotating face

- ✓ **The polar coordinate system translates rotation into displacement along the θ -axis.**
- \+ **This transformation allows for uniform representation of rotated faces through the translation equivariance of convolutional neural networks.**



Architecture: RP-Net for Rotation-Invariant Face Detection

- ✓ Initial stages use a lightweight network for face region extraction
- ✓ Subsequent stages : apply Polar transformation for orientation normalization.
- ✓ Final output: includes landmark localization with ASCR strategy for precise facial mapping.

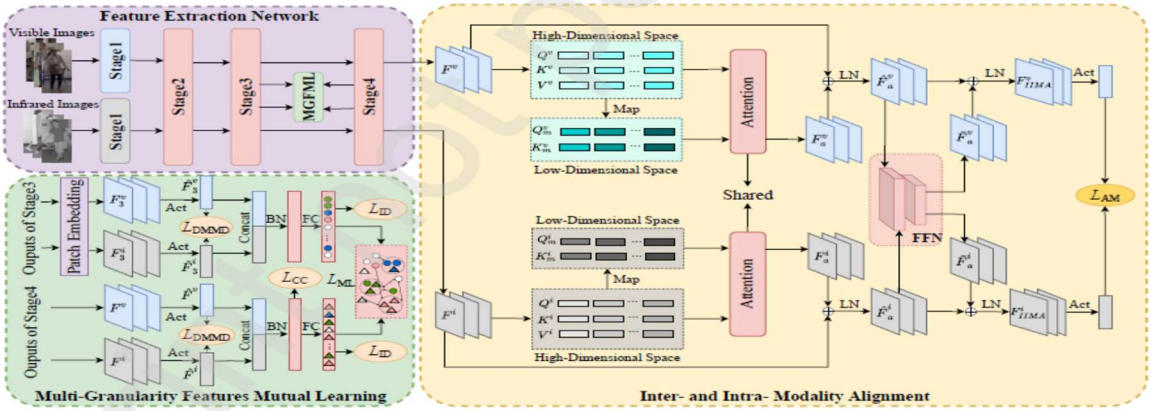


Flowchart. ASCR method

- ✓ Shows how the PSS and PCS submodules methodically assign differential weights to the spatial and channel dimensions of the polar feature map V to refine the precision of face alignment

Comparison of face detection performance in a challenging rotational scenario using (a) MTCNN, (b) PCN, and (c) RP-Net

Transformer Net	2025-94	



Overall architecture. DMANet method for VI-ReID.

- ✓ 'MGFML' denotes the multigranularity features mutual learning module.
- ✓ During the test, we only retain baseline model and MGFML module after Stage 4 to achieve the cross-modality retrieval
- o DMANet: Dual-Modality Alignment Network for Visible-Infrared Person Re-Identification

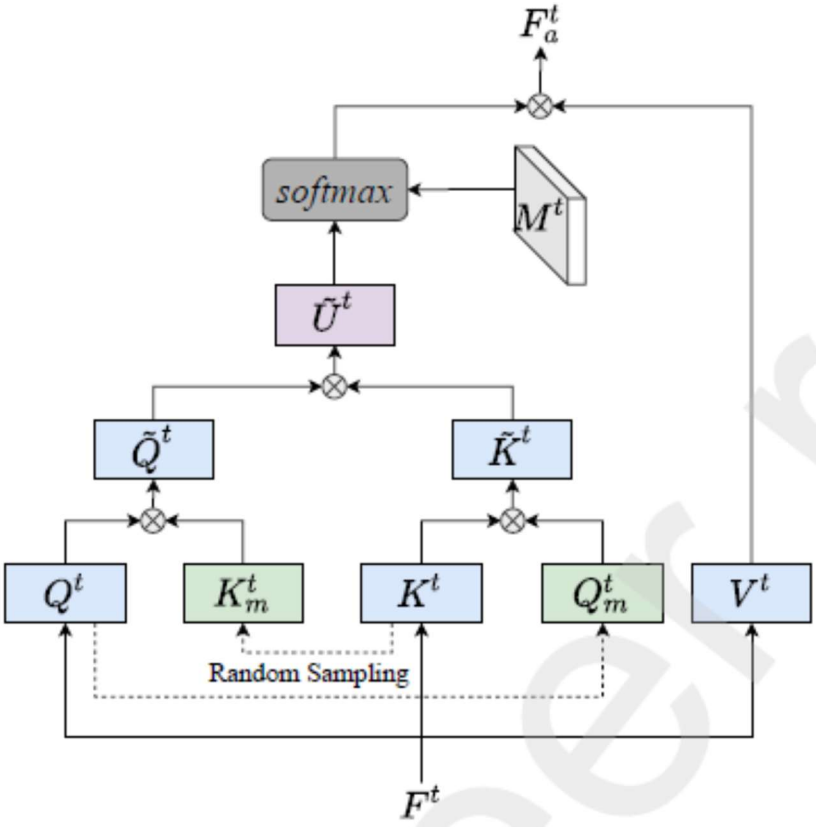
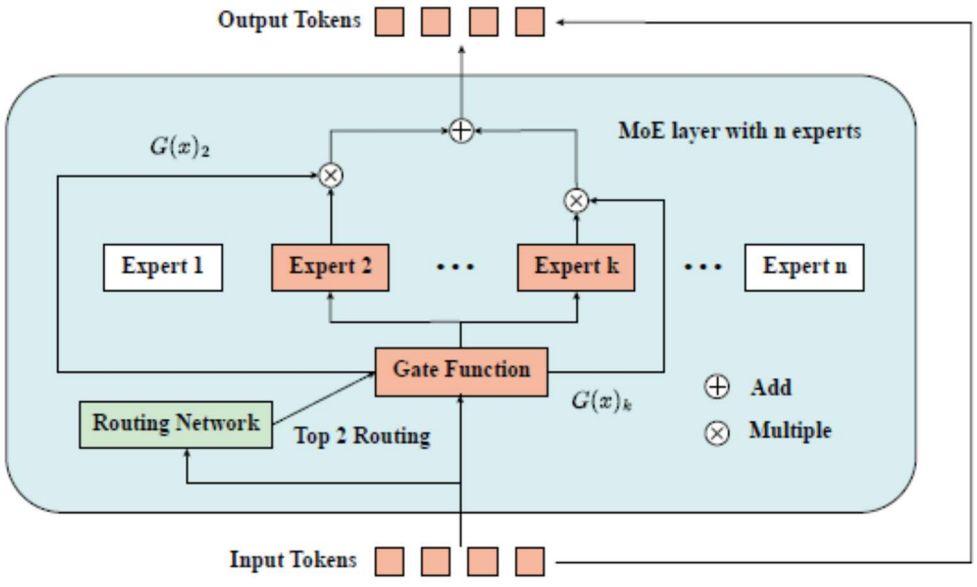
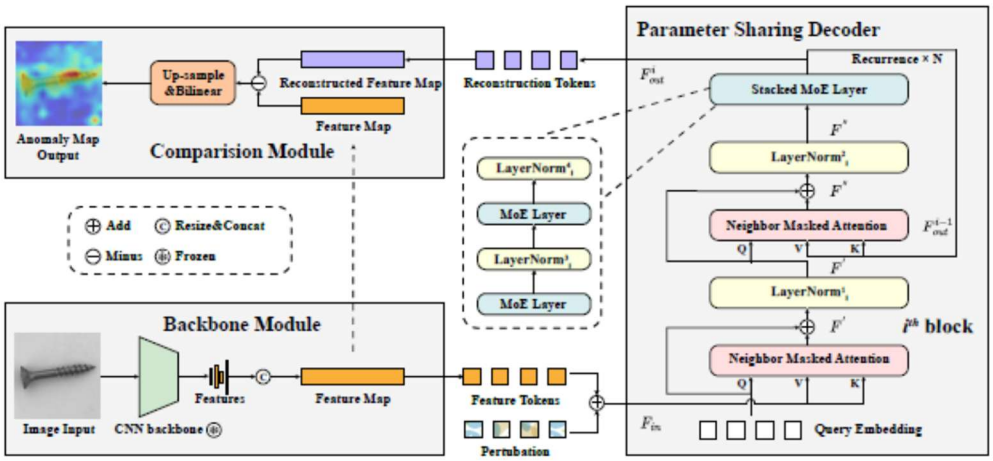


Illustration of Attention in visible branch of IIMA

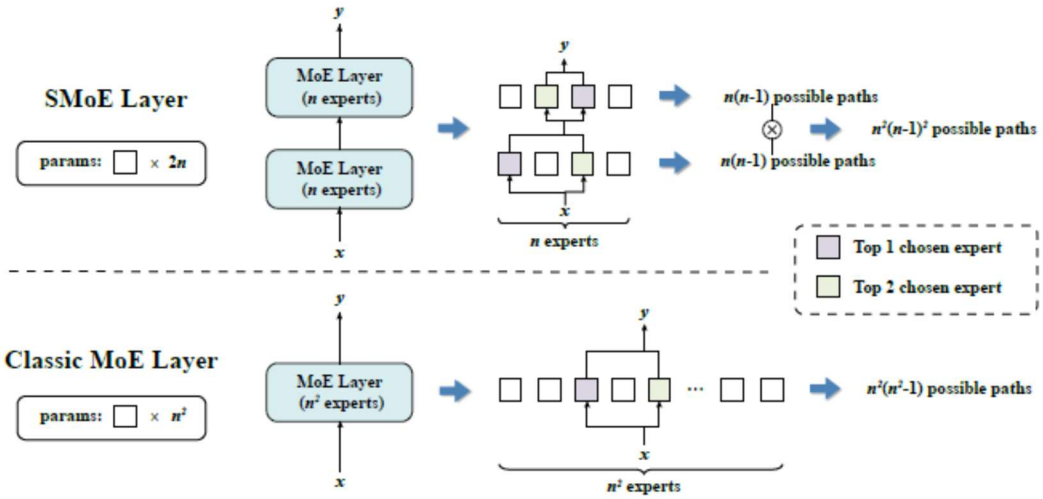
Transformer Net	2025-95	



Architecture. MoE layer used in MoEAD



Architecture. framework. MoEAD



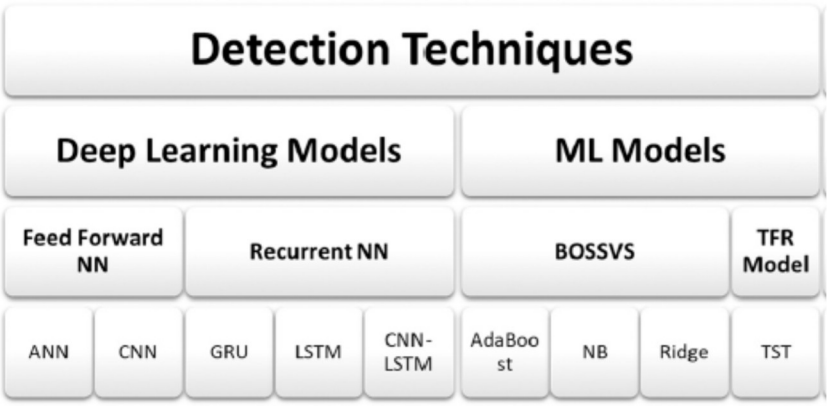
SMoE layer compared with the classic MoE layer.

+ Two stacked MoE layers
  enhance the selectable paths through multiplication, thus strengthening the gating selection ability
+ Compared to a layer with a larger number of experts, this approach achieves a similar effect with fewer trainable parameters

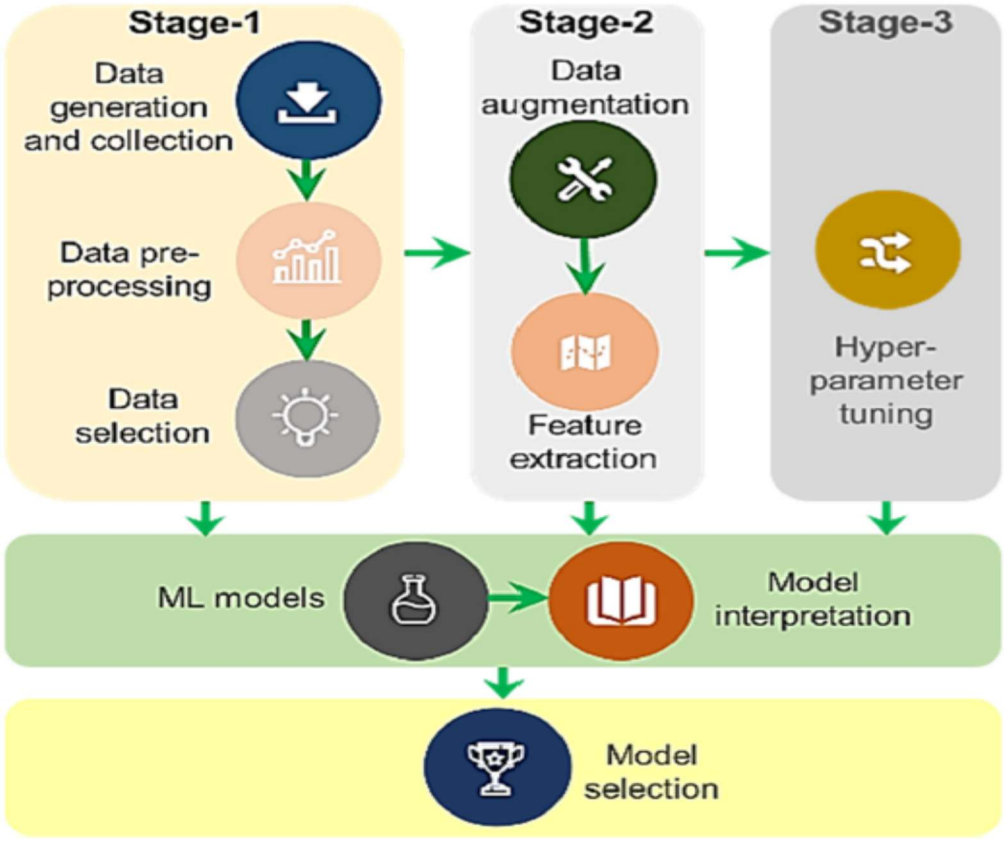
(a) Ablation studies on architecture design.

MoE	Stacked	Att-Shared	MoE-Shared	Params	FLOPs	I-AUROC	P-AUROC
				4.6M	2.16G	96.1	96.3
		✓		2.8M	2.16G	95.7	96.1
✓				10.9M	2.17G	97.4	96.9
✓		✓		9.1M	2.17G	97.2	96.8
✓			✓	4.6M	2.17G	97.1	96.8
✓		✓	✓	2.8M	2.17G	97.1	96.8
✓	✓			19.3M	2.18G	98.1	97.2
✓	✓	✓		17.5M	2.18G	97.9	97.1
✓	✓		✓	6.7M	2.18G	97.7	97.0
✓	✓	✓	✓	4.9M	2.18G	97.7	97.0

Transformer Net	2025-96	



Implemented Machine and Deep Learning models for Cyberattack detection



Framework. Classification

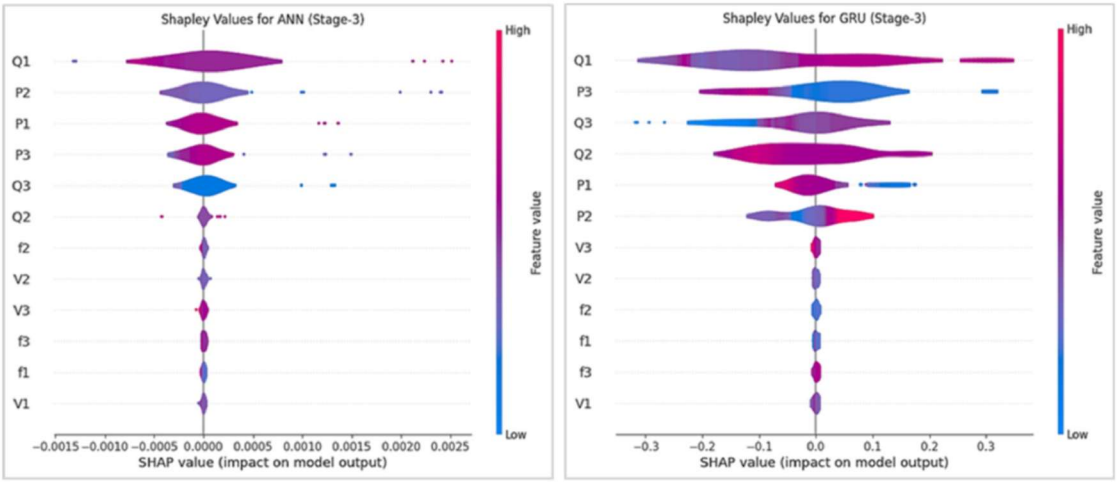


Fig. 13. SHAP plot for feature importance at Stage-3.

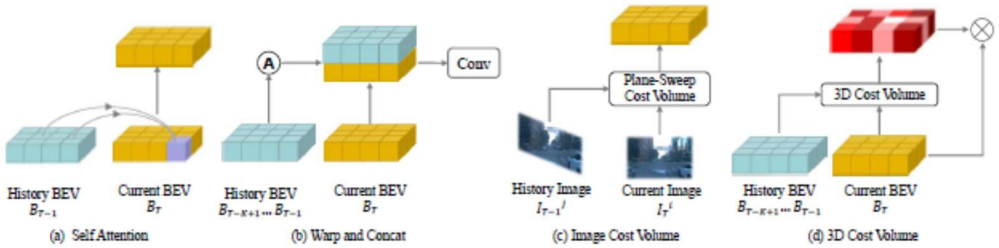


Fig. 1: Comparison of Temporal Fusion Methods. Illustrated are four key approaches: (1) Temporal Self-Attention [20], leveraging attention mechanisms for temporal integration; (2) Warp and Concat [8,35,37], combining features across frames and fusing them through convolution; (3) Cost Volume Construction in image space [25], constructing cost volume from image input of different frames and leveraging plane-sweep volumes for depth map generation; and (4) Our Proposed Method, which involves constructing a temporal cost volume in 3D space to enhance feature refinement. In the figure, Ⓐ and ⊗ represent coordinate alignment and element-wise product, accordingly.

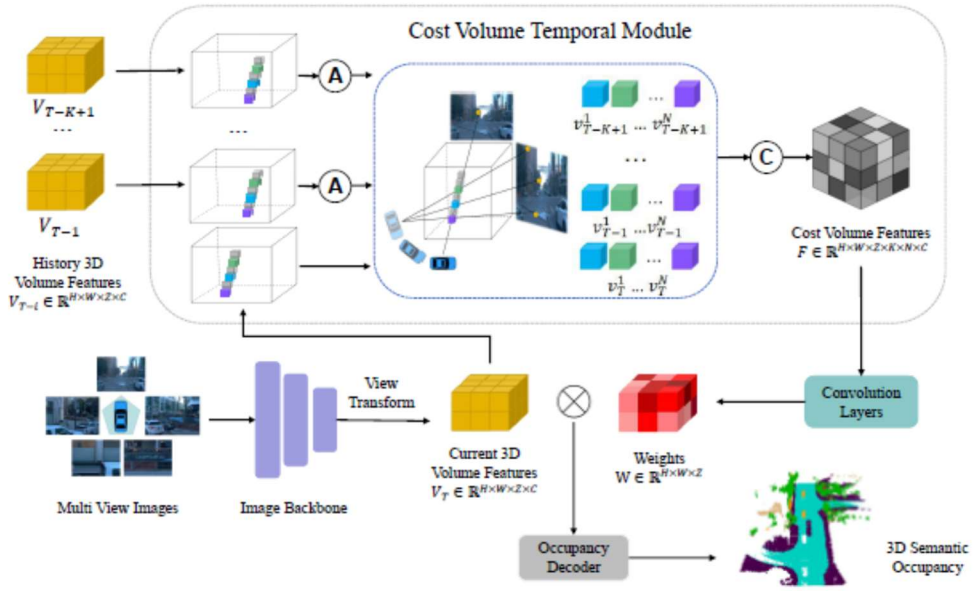
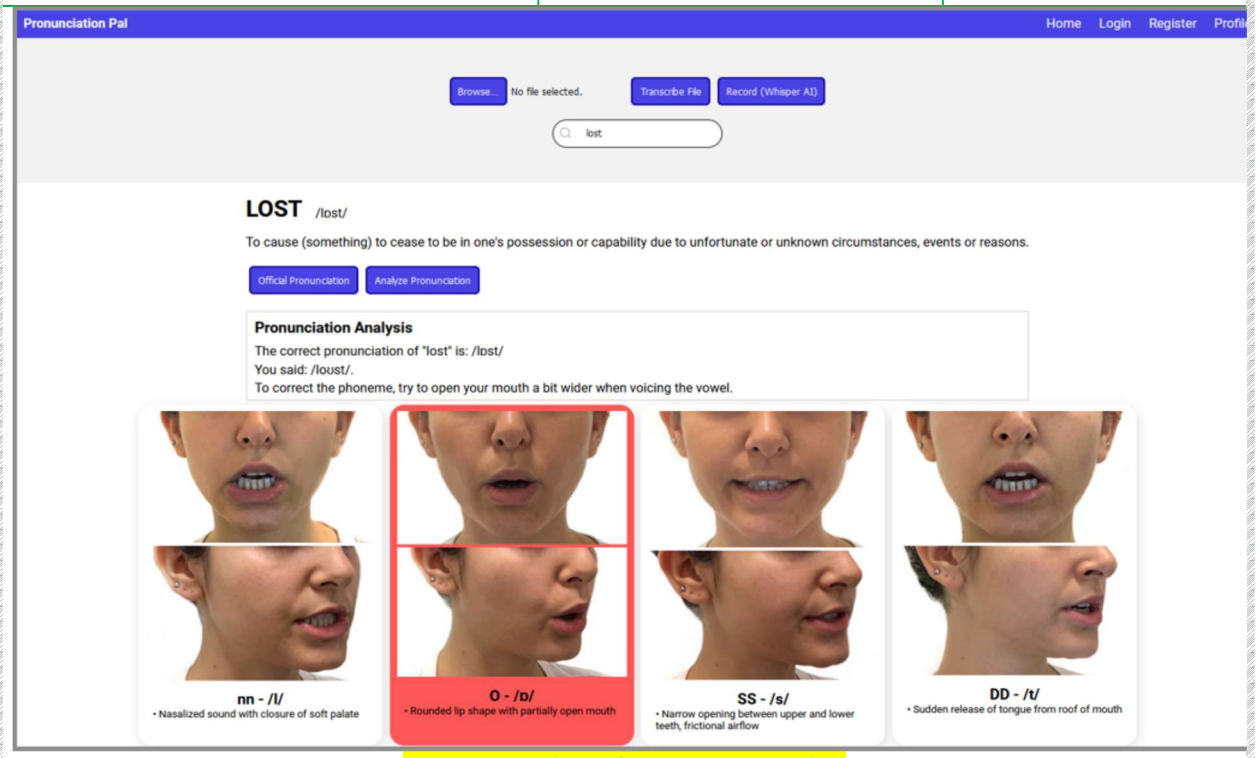


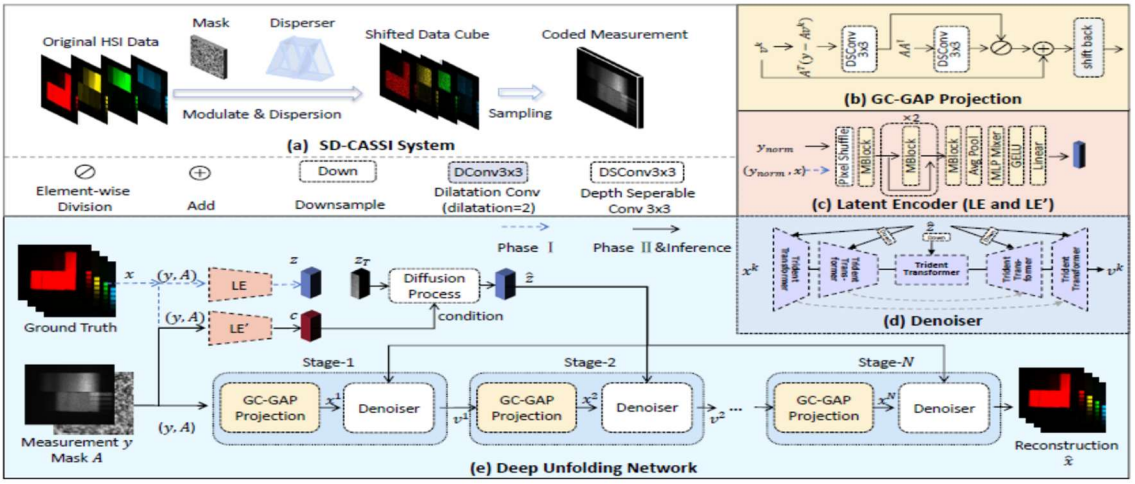
Fig. 2: Overall Architecture of CVT-Occ. The image backbone extracts multi-scale features from multi-view images, which are transformed into 3D volume features denoted as $\mathbf{V} \in \mathbb{R}^{H \times W \times Z \times C}$. The Cost Volume Temporal Module samples points along the line of sight within the current volume and projects them onto $K-1$ historical frames, resulting in $K \times N$ 3D volume features. These features are concatenated to construct cost volume features $\mathbf{F} \in \mathbb{R}^{H \times W \times Z \times (K \times N) \times C}$. Convolution layers are then applied to generate weights $\mathbf{W} \in \mathbb{R}^{H \times W \times Z}$, refining the depth of 3D voxel. Finally, an occupancy decoder produces 3D semantic occupancy predictions. In the figure, Ⓐ, Ⓒ, and ⊗ represent coordinate alignment, concatenation, and element-wise product, respectively.

Transformer Net	2025-98	

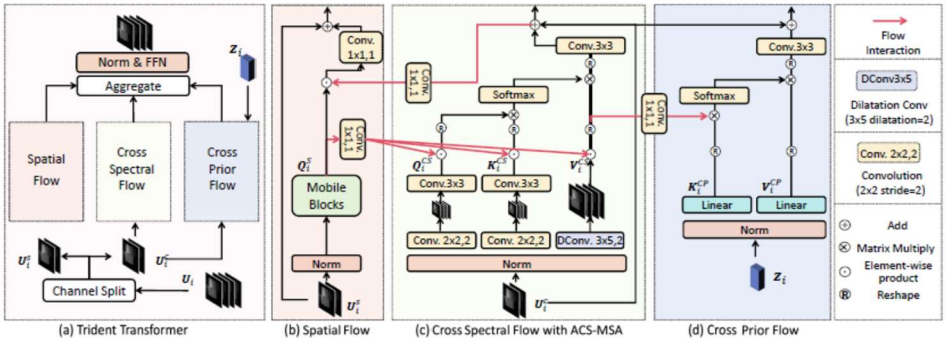


Demonstration of Feedback Engine

Transformer Net	2025-100	



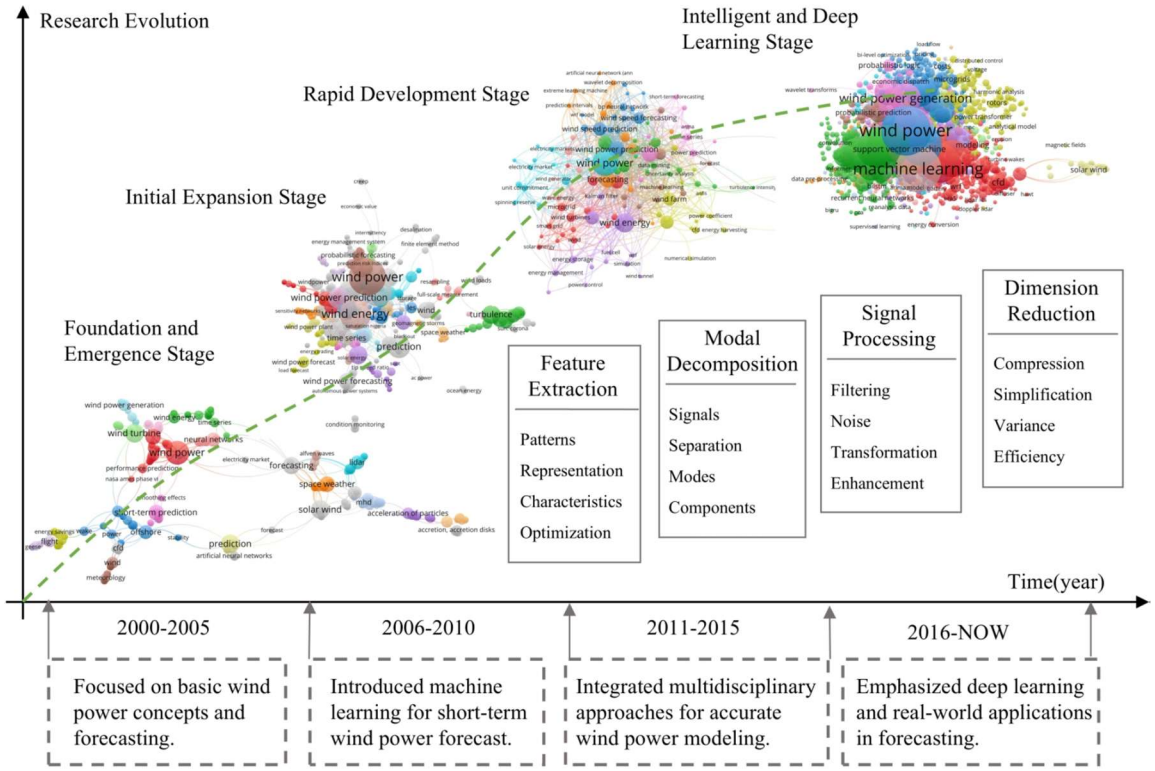
(a) The single disperser CASSI imaging process. HSI data cube is captured by a monochromatic sensor.
(b) GC-GAP projection.
(c) Latent encoder.
(d) Simplified Denoiser.
(e) The measurement y and masks A pass through an N-stage DUN, where each stage is composed of a GC-GAP projection and a denoiser.
The denoiser follows a U-shape structure and consists of five Trident Transformers (TT), where each TT is assisted with prior knowledge zGT generated from the diffusion model



Trident Transformer in Fig. 3(d).

- ✓ **(b)-(d): detailed sub-modules. Ui is the input feature. The prior feature Zi is sent into the prior flow**

Transformer Net	2025-17	



Evolution map of Wind power forecasting research

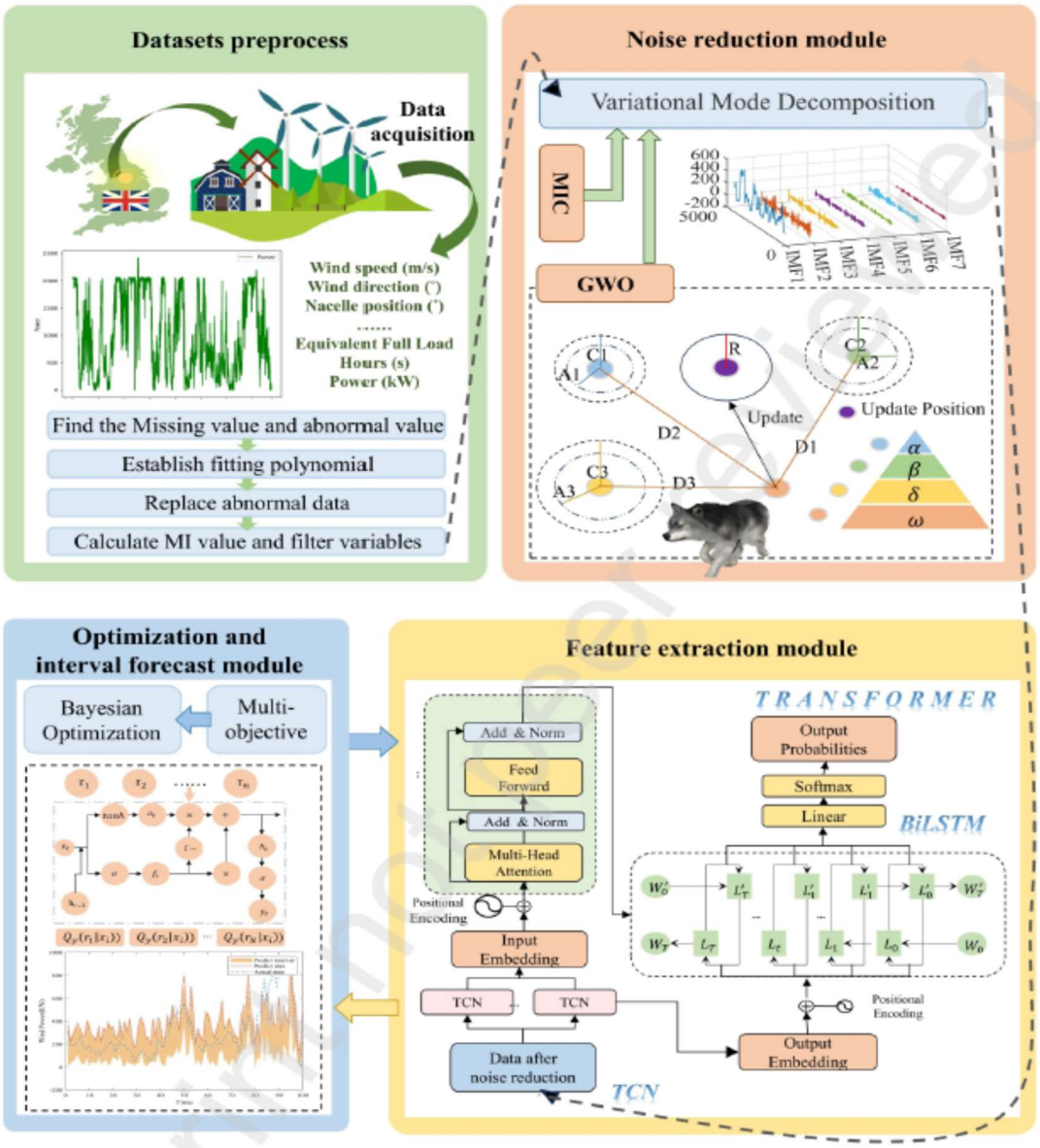


Fig.3 IVMD-MOBO-QRTBT model framework

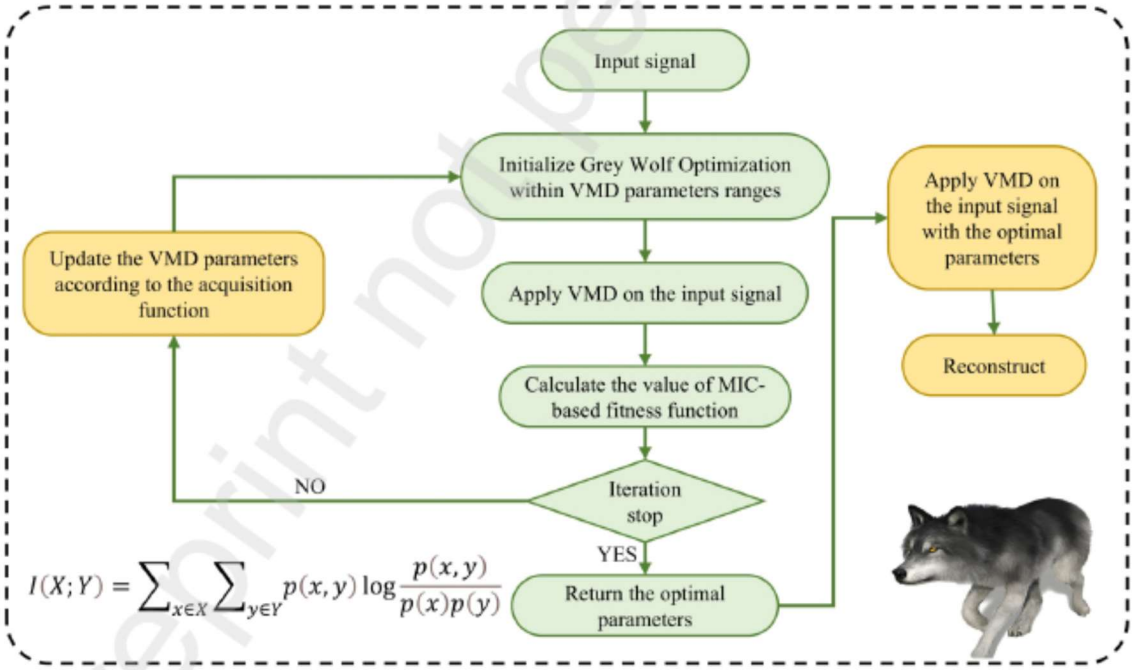


Fig.4 The process of noise reduction module

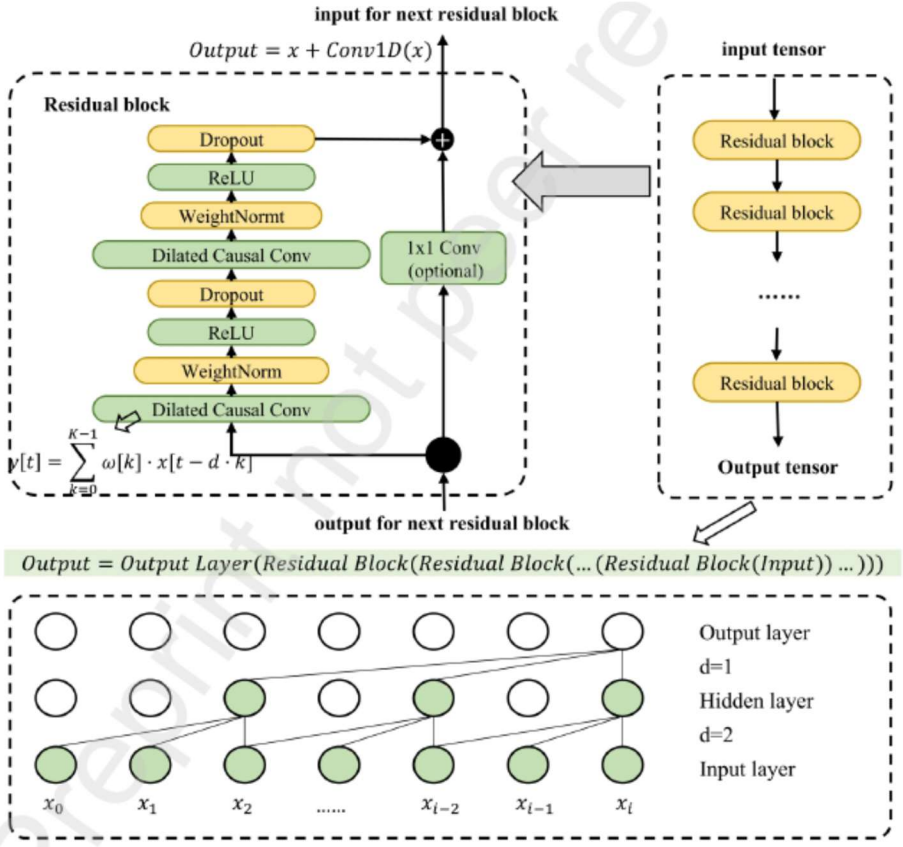


Fig.5 TCN network structure

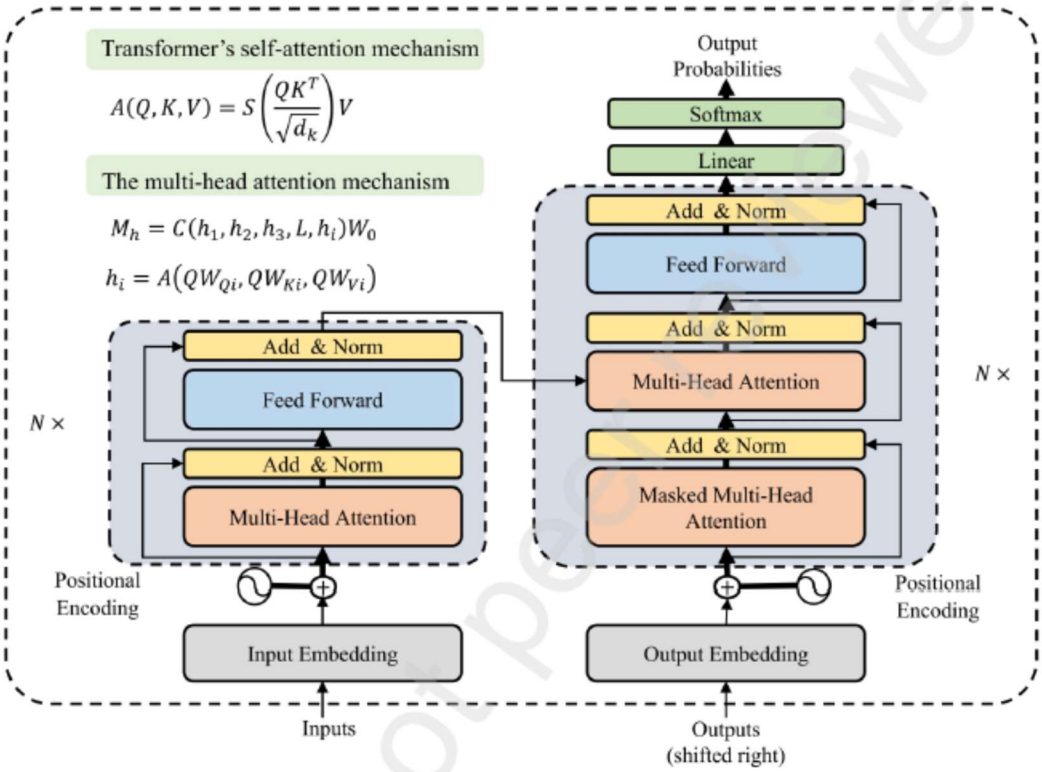


Fig.6 Transformer network structure diagram

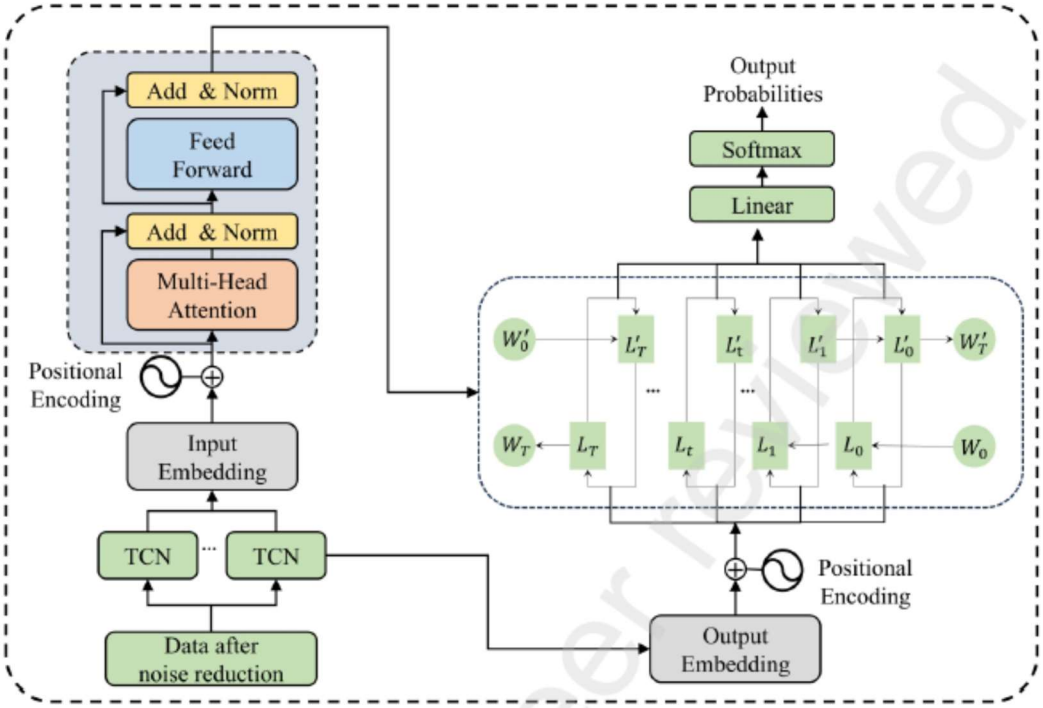
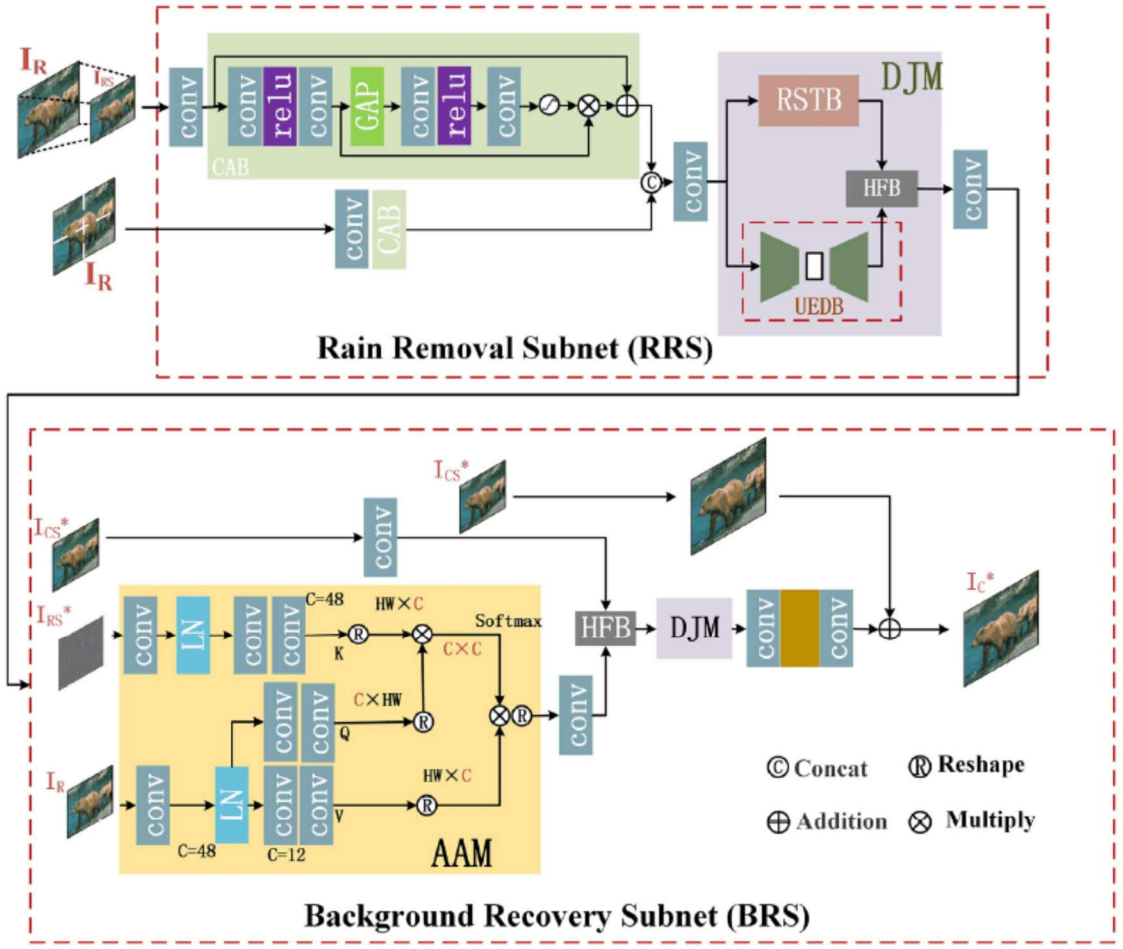


Fig.7 The structure diagram of feature extraction module

Table 8 The results of evaluation indicators in each season

Season	Model	*MAE*	*MSE*	*RMSE*	R^2	*PICP*	*PINAW*	*WS*
Spring	IVMD-BO-QRB	21.12	1079.12	32.85	0.73	56.18%	15.51%	47.47%
	IVMD-BO-QRTT	21.22	1022.08	31.97	0.74	31.46%	46.88%	16.71%
	IVMD-BO-QRTLT	21.67	1089.00	33.00	0.74	51.69%	31.15%	35.59%
	IVMD-BO-QRTBT	21.79	1112.22	33.35	0.74	83.15%	17.99%	68.19%
	IVMD-MOBO-QRB	20.69	1090.98	33.03	0.74	92.14%	12.94%	80.22%
	IVMD-MOBO-QRTT	21.53	1011.24	31.80	0.74	96.63%	18.21%	79.03%
	IVMD-MOBO-QRTLT	19.72	996.03	31.56	0.76	91.91%	14.56%	78.53%
	IVMD-MOBO-QRTBT	19.98	1060.80	32.57	0.76	96.40%	15.11%	81.83%

Transformer Net	2025-20	



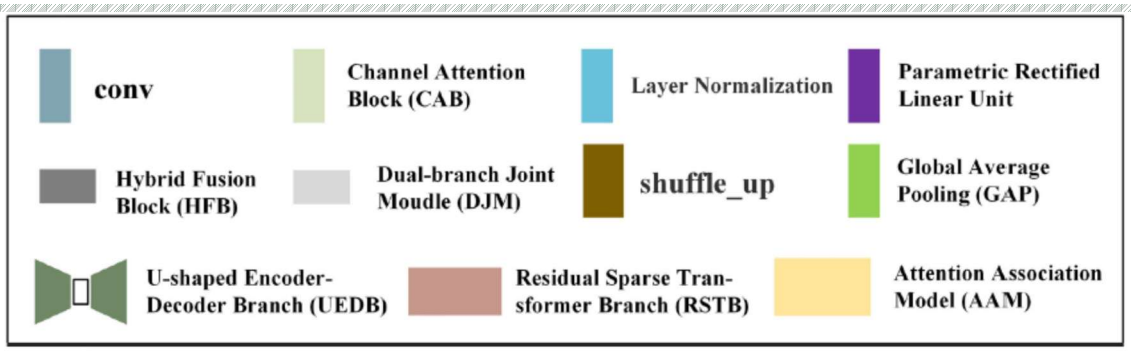
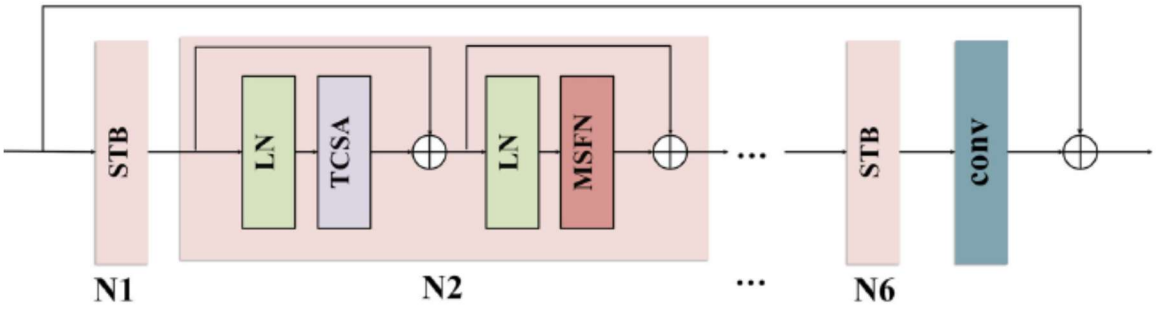
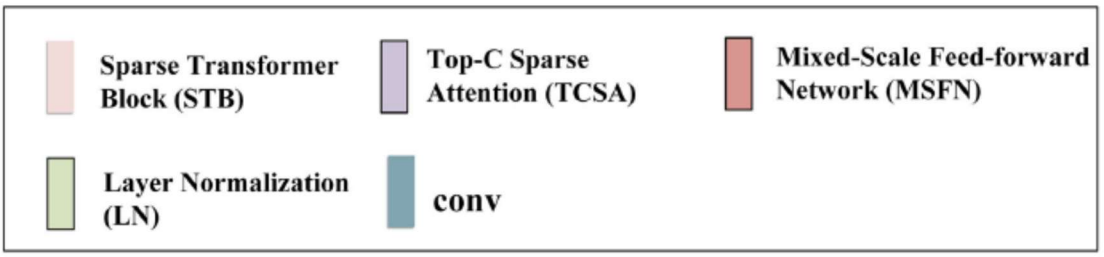
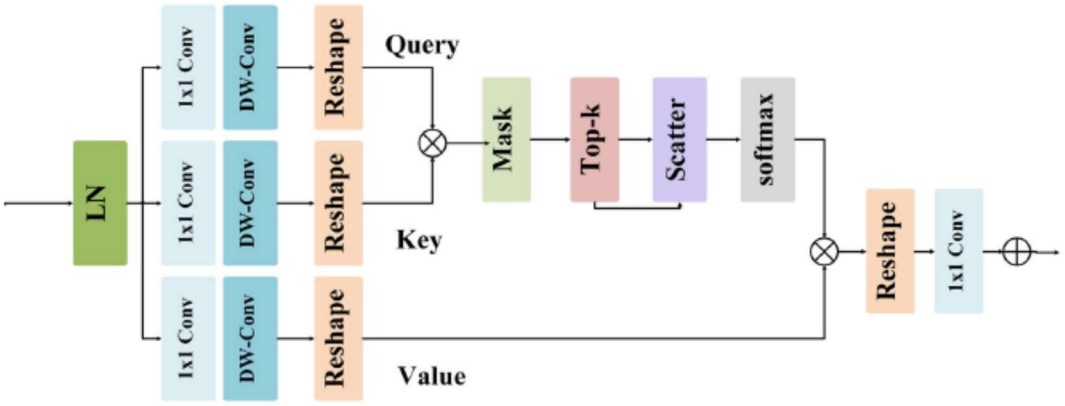


Fig. 1 Proposed network structure

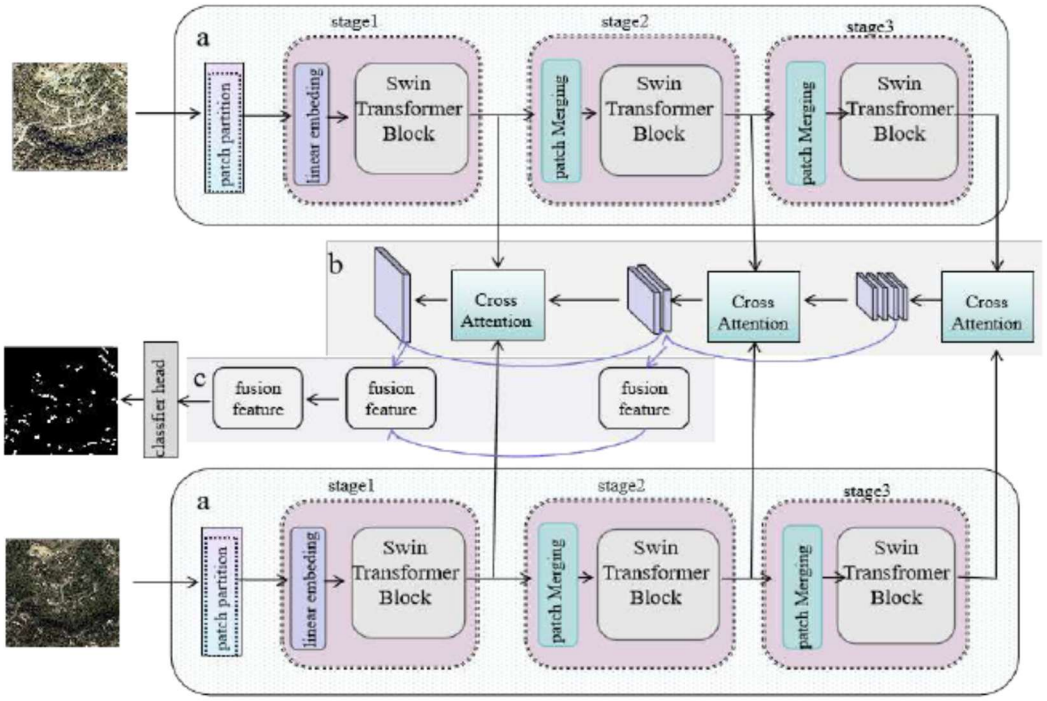




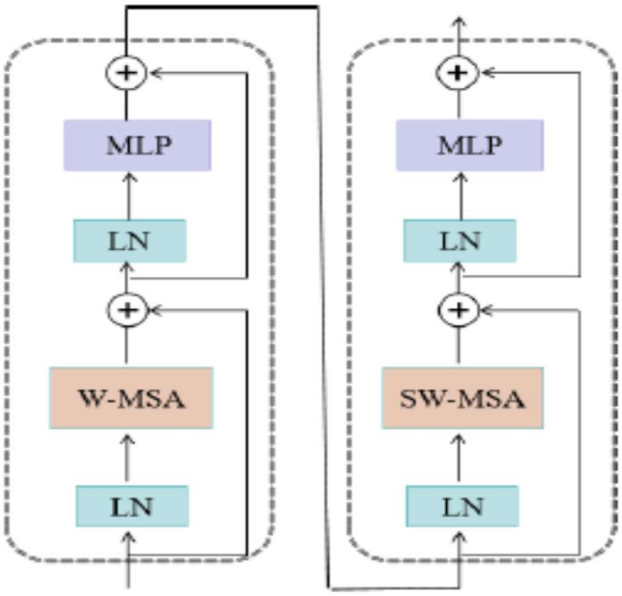
(a) Residual Sparse Transformer Branch (RSTB)



(b) Top-C Sparse Attention (TCSA)



Overall structure of the SCACD-Net. (a) Feature extraction network. (b) Cross-attention difference feature extraction network. (c) Cascaded feature fusion network.



Swin Transforemr block.

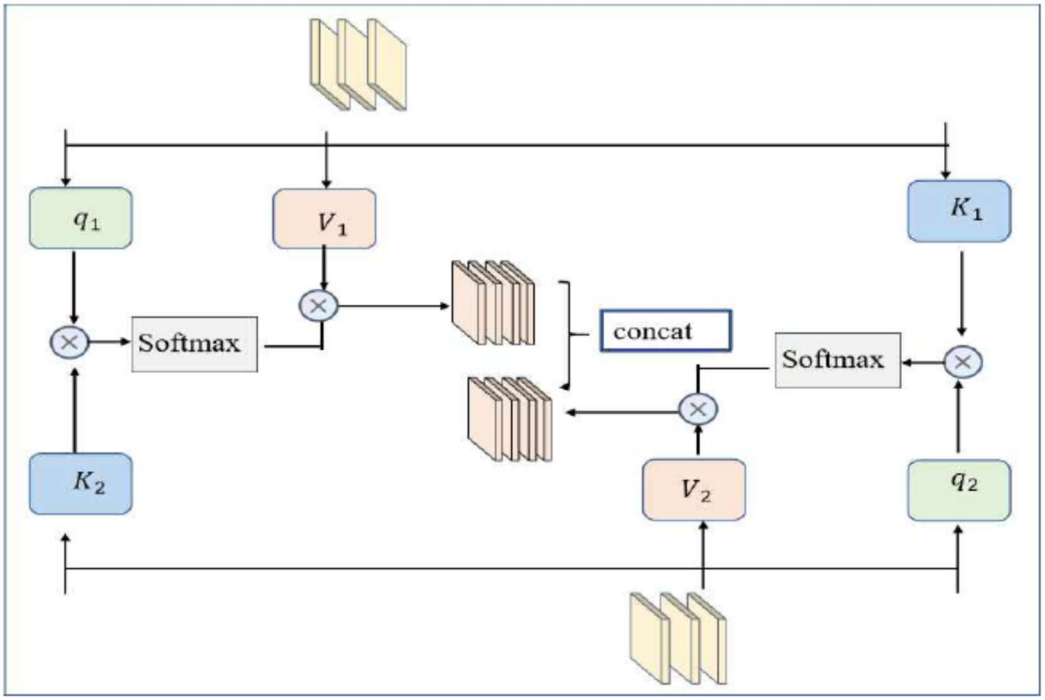


Fig. 3 Cross-Attention difference Feature Extraction Module.

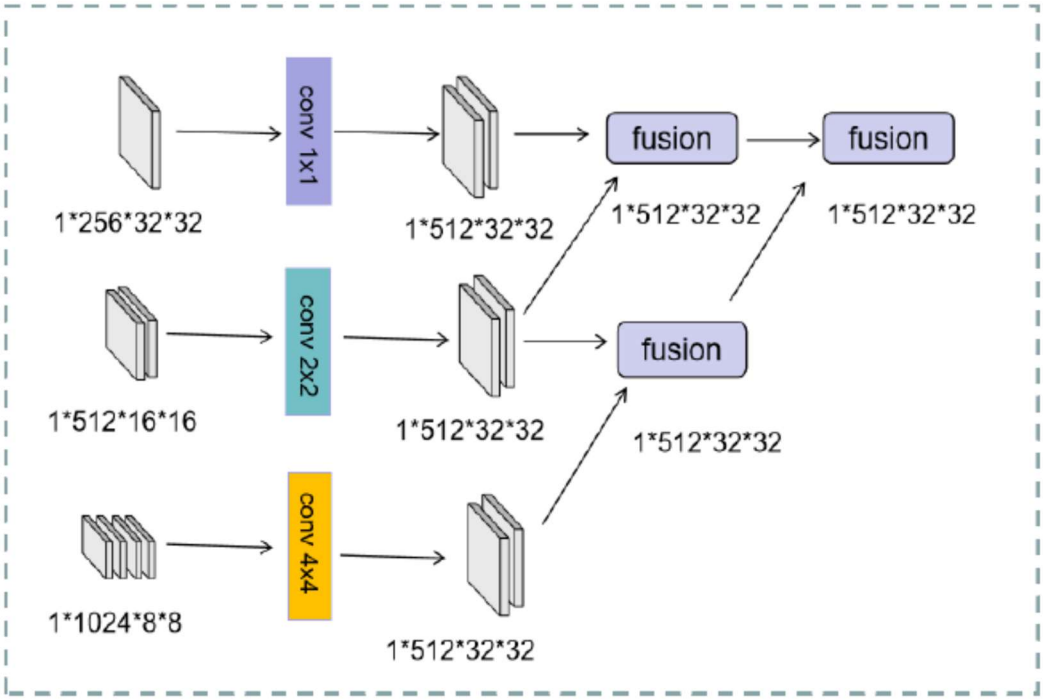
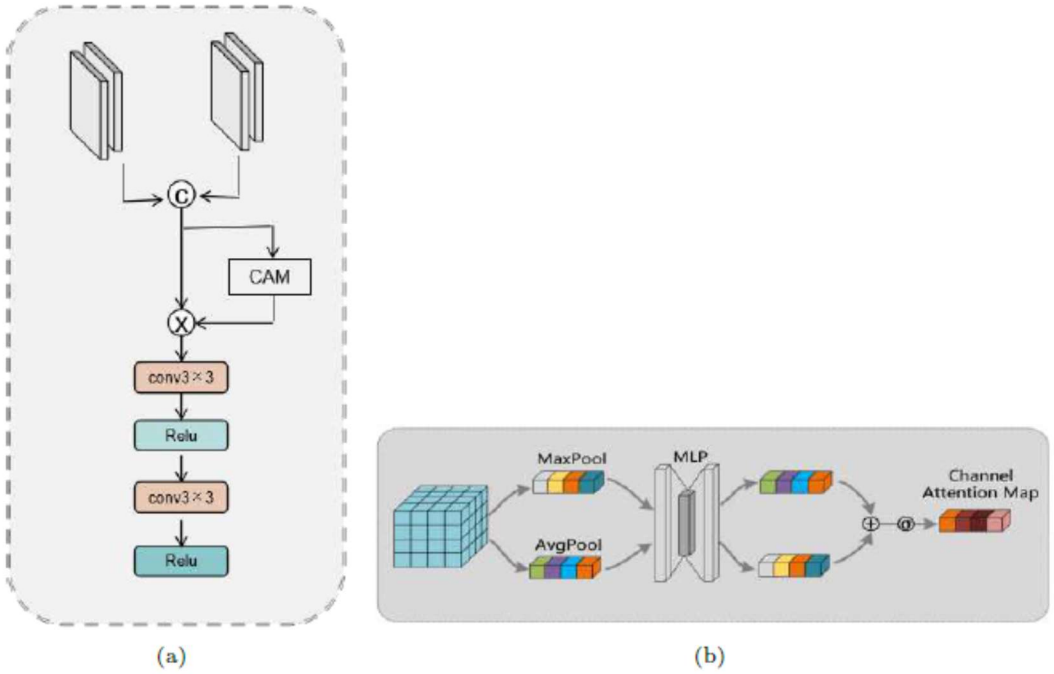


Fig. 4 Cascade feature fusion network.



(a) (b)

Fig. 5 (a) Feature Fusion block. (b) Channel Attention Mechanism (CAM)