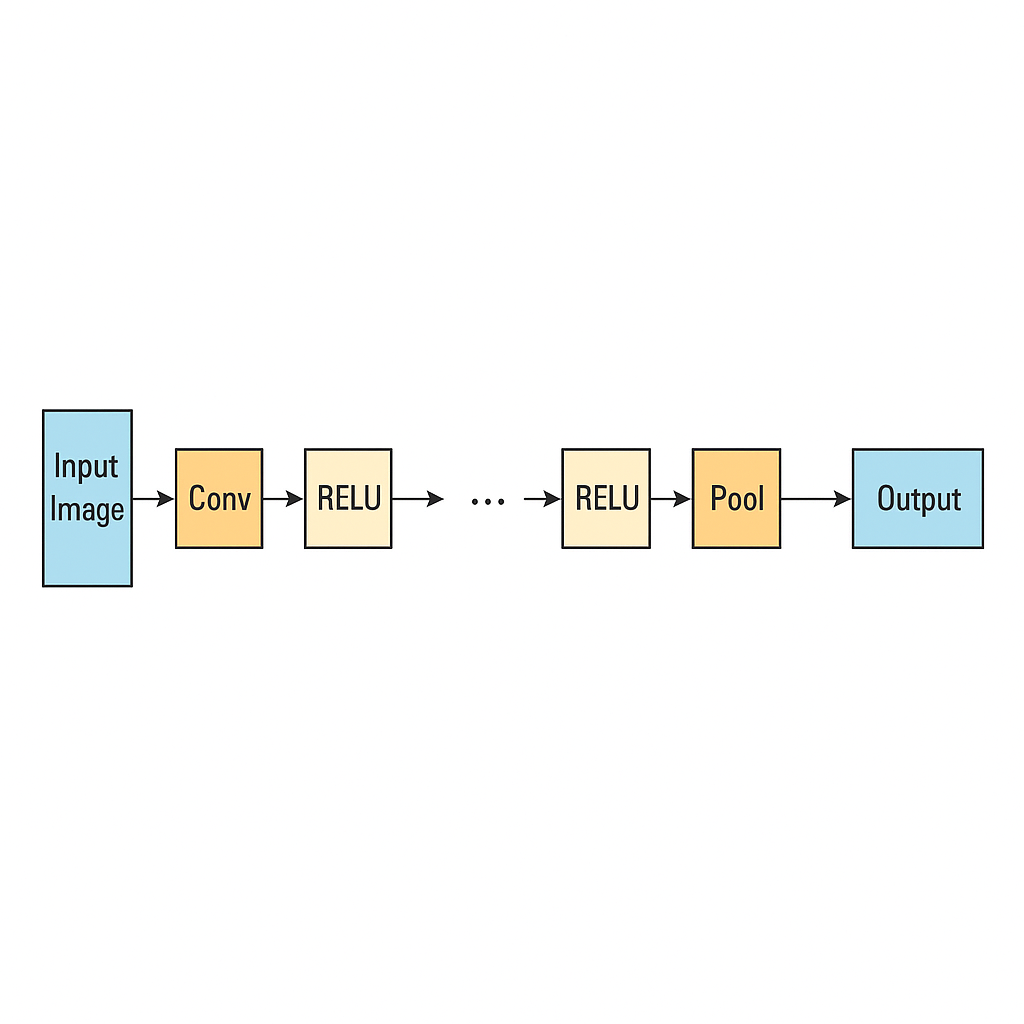
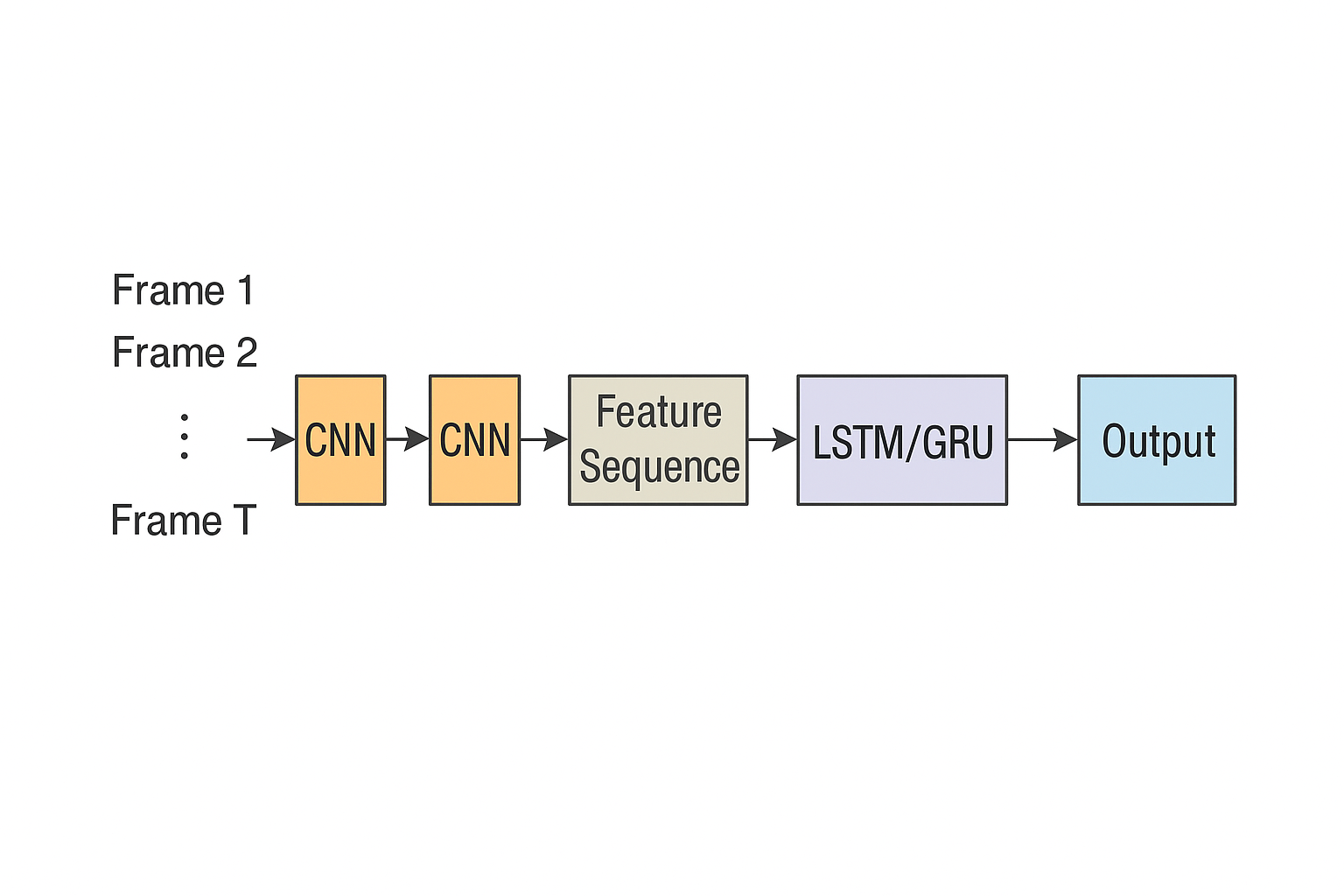
White Paper on Image and Video AI Models

# 2. Image AI Models

This section covers the major deep learning models used for image-based tasks. Each model type is briefly explained along with its architecture, use cases, and popular tools.

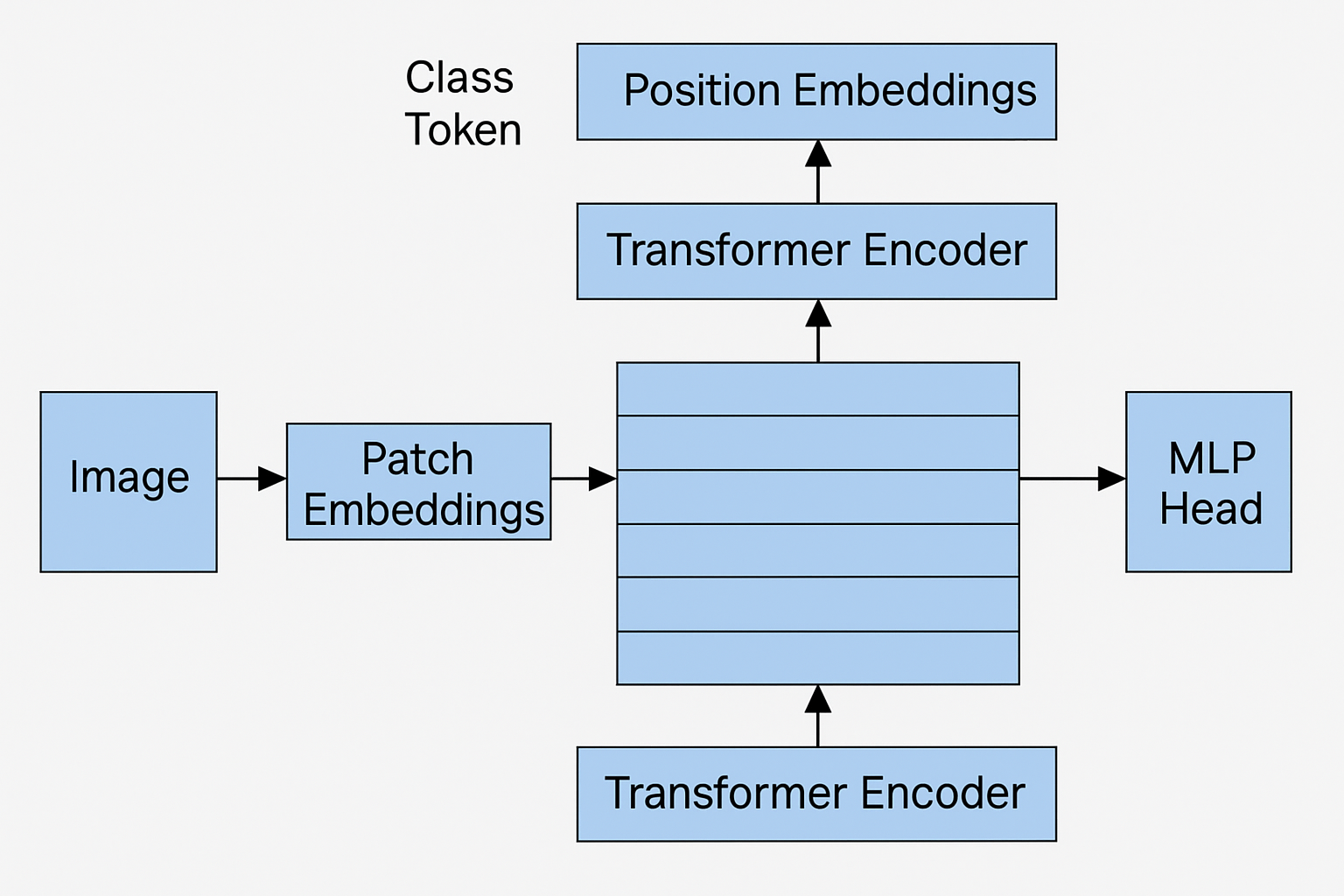
## 2.1 Convolutional Neural Networks (CNNs)

Brief Description:  
CNNs are the foundational models for image tasks. They use local filters (kernels) to extract hierarchical spatial features.  
  
Architecture:  
Input Image → [Conv → ReLU → Pool]\* → Flatten → FC → Output  
  


Use Cases:  
- Image classification (e.g., ResNet, VGG)  
- Object detection (e.g., YOLO, SSD)  
- Semantic segmentation (e.g., U-Net)  
  
Tools/Frameworks:  
- PyTorch, TensorFlow, Keras  
- OpenCV, torchvision, timm

## 2.2 Vision Transformers (ViT)

Brief Description:  
ViTs treat an image as a sequence of patches and apply standard Transformer attention over these patches.  
  
Architecture:  
Image → Patch Embedding → Positional Encoding → [Transformer Block]\* → MLP → Output

  
  
Use Cases:  
- Image classification  
- Fine-grained recognition  
- Zero-shot learning (e.g., CLIP)  
  
Tools/Frameworks:  
- HuggingFace transformers, timm, DINOv2

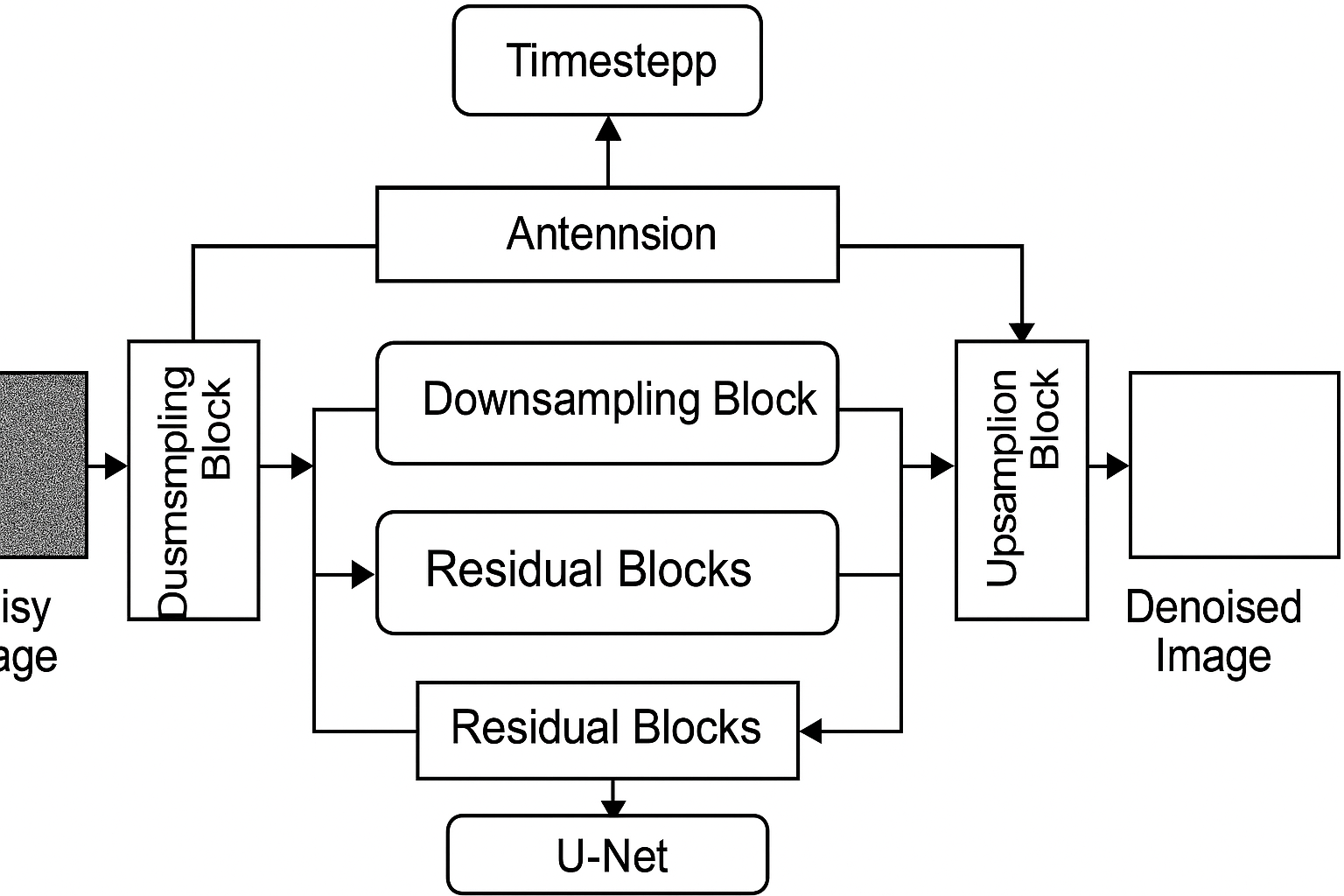
## 2.3 Swin Transformer

Brief Description:  
A hierarchical Transformer using shifted windows to perform attention in local and global contexts.  
  
Architecture:  
Image → Patch Partition → Window Attention → Merge & Shift → MLP → Patch Merge → ...  
  
Use Cases:  
- Super-resolution (SwinIR)  
- Image segmentation (Swin-UNet)  
- Restoration and denoising  
  
Tools/Frameworks:  
- SwinIR, MMSegmentation, timm

## 2.4 GANs (Generative Adversarial Networks)

Brief Description:  
GANs consist of a generator that creates images, and a discriminator that tries to distinguish fake from real.  
  
Architecture:  
Generator: z → Deconv Layers → Image  
Discriminator: Image → Conv Layers → Real/Fake  
  
Use Cases:  
- Image synthesis (e.g., StyleGAN)  
- Super-resolution (e.g., Real-ESRGAN)  
- Image-to-image translation (e.g., Pix2Pix)  
  
Tools/Frameworks:  
- Real-ESRGAN, StyleGAN, NVIDIA StyleGAN3

## 2.5 Diffusion Models

Brief Description:  
Diffusion models reverse a gradual noising process to generate high-quality images.  
  
Architecture:  
U-Net + Timestep Embeddings + Attention  
  
Use Cases:  
- Text-to-image (Stable Diffusion)  
- Super-resolution (StableSR)  
- Inpainting and editing  
  
Tools/Frameworks:  
- HuggingFace Diffusers, CompVis, OpenAI Glide

## 2.6 Autoencoders (AE & VQ-VAE)

Brief Description:  
Autoencoders compress and reconstruct images. VQ-VAE adds quantization for discrete representations.  
  
Architecture:  
Encoder: Image → Latent Code  
Decoder: Latent → Reconstructed Image  
  
Use Cases:  
- Image compression  
- Denoising  
- Latent code manipulation  
  
Tools/Frameworks:  
- OpenAI DALL·E VQ-VAE, PyTorch AEs, TensorFlow AE

# 3. Video AI Models

Video AI models extend image models to handle temporal information across frames. They capture motion, context, and patterns over time using various architectural strategies.

## 3.1 3D Convolutional Neural Networks (3D CNNs)

3D CNNs apply convolution operations over spatial and temporal dimensions. They are well-suited for capturing motion-aware features.  
  
Architecture:  
Input: Video Clip → 3D Conv → 3D Pool → FC → Output  
  
Use Cases:  
- Action recognition  
- Gesture recognition  
- Video classification  
  
Tools/Frameworks:  
- PyTorchVideo  
- MMAction2  
- Kinetics Dataset

# 3.2 CNN + RNN/LSTM Hybrid

**Brief Description**  
Combines CNNs (for spatial features) with RNNs or LSTMs (for temporal dynamics), often using a frame-by-frame CNN followed by a sequence model.

**Architecture**

text

CopyEdit

Each Frame → CNN → Feature Sequence → LSTM/GRU → Output

**Use Cases**

* Video classification with temporal memory
* Human activity analysis
* Behavior prediction

**Tools/Frameworks**

* Keras, PyTorch
* TensorFlow Seq2Seq APIs
* OpenPose + LSTM (for pose tracking)

# 3.3 TimeSformer

**Brief Description**  
A pure Transformer model for video understanding using **space-time attention**. It applies self-attention across both spatial and temporal dimensions.

**Architecture**

text

CopyEdit

Video → Patchify → Spatial Transformer → Temporal Transformer → MLP → Output

**Use Cases**

* Action recognition
* Event detection
* Temporal video tagging

**Tools/Frameworks**

* Facebook TimeSformer repo
* HuggingFace Video Transformers (coming)
* PyTorchVideo

# 3.4 Video Swin Transformer

**Brief Description**  
A hierarchical attention-based model adapted from Swin Transformer for efficient long-video modeling via **shifted 3D windows**.

**Architecture**

text

CopyEdit

Video → 3D Patch Partition → Window Attention → Merge → Downsample → Output

**Use Cases**

* Action recognition
* Fine-grained video understanding
* Video question answering

**Tools/Frameworks**

* MMAction2 (Swin backend)
* HuggingFace (Swin video variants)

# 3.5 Video GANs

**Brief Description**  
Extends GANs to model temporal consistency across frames. Includes spatio-temporal convolutions or recurrent mechanisms in the generator.

**Architecture**

text

CopyEdit

Generator: z → Conv3D / Frame-wise Conv + LSTM → Video Sequence

Discriminator: Sequence → Conv3D or LSTM-based classifier

**Use Cases**

* Video super-resolution (e.g., TecoGAN)
* Synthetic video generation
* Deepfakes (when combined with face GANs)

**Tools/Frameworks**

* TecoGAN (video SR)
* MoCoGAN / VGAN
* DeepFake libraries (FaceSwapGAN)

3.6 Video Diffusion Models

**Brief Description**  
Applies diffusion techniques across video frames to generate realistic, temporally consistent video. Computationally heavy, but cutting edge.

**Architecture**

text

CopyEdit

Latent Video Encoding → Reverse Denoising Process

(uses 3D U-Net or frame-conditioned models)

**Use Cases**

* Text-to-video (e.g., Runway Gen-2, Sora)
* Video inpainting or editing
* Creative storytelling

**Tools/Frameworks**

* ModelScope video diffusion
* Runway Gen-2 (closed source)
* SVD (Stable Video Diffusion by StabilityAI)

# 3.7 Optical Flow-Based Models

**Brief Description**  
Uses optical flow to model pixel-level motion between frames. Can be standalone or used as part of other networks.

**Architecture**

text

CopyEdit

Frame t and t+1 → FlowNet / RAFT → Motion Field → Warping / Prediction

**Use Cases**

* Video interpolation
* Motion tracking
* Scene flow analysis

**Tools/Frameworks**

* RAFT (Recurrent All-Pairs Field Transforms)
* FlowNet2.0
* DAIN (Depth-Aware Interpolation Network)

Figure: Diagram of various video AI model architectures.