**Appendix: Keywords and Definitions**

Below is a list of key technical terms used in the proposed GIF Enhancement Pipeline and the integrated GFPGAN framework, with precise definitions:

**A. Input and Preprocessing**

**🔑 GIF (Graphics Interchange Format)**  
A bitmap image format supporting animation by combining multiple frames in a single file. In this project, it serves as the degraded low-resolution input.

**🔑 Frame Extraction**  
The process of splitting an animated GIF into individual image frames for per-frame processing. This allows applying restoration and upscaling to each frame independently.

**🔑 PIL (Python Imaging Library)**  
A Python library for opening, manipulating, and saving image files. Used here for reading GIFs, extracting frames, and saving enhanced images.

**B. Face Restoration (GFPGAN Base)**

**🔑 GFPGAN (Generative Facial Prior GAN)**  
A generative adversarial network designed for blind face restoration. It uses a generative facial prior to recover realistic facial details from degraded inputs.

**🔑 U-Net**  
A convolutional neural network architecture with encoder-decoder structure and skip connections. In GFPGAN, it acts as a *degradation removal module* that cleans the input image before restoration.

**🔑 Latent Code Mapping (MLP)**  
A multi-layer perceptron (MLP) that converts the output features from the U-Net into a latent vector (w) used by the generative prior (StyleGAN2).

**🔑 StyleGAN2**  
A state-of-the-art generative adversarial network that acts as a *prior*. It generates realistic high-resolution facial textures conditioned by the latent code.

**🔑 CS-SFT (Channel-Split Spatial Feature Transform)**  
A modulation layer that injects spatial features from the U-Net into the StyleGAN generator. It allows controlling generated features at different spatial locations to preserve structure and identity.

**🔑 Facial Component Loss**  
A loss function that ensures fine-grained facial regions (like eyes, mouth) are realistically restored. Often includes region-specific discriminators.

**🔑 Identity Loss**  
A loss term ensuring that the restored face maintains the same identity as the original degraded input.

**C. Background Enhancement**

**🔑 Real-ESRGAN (Real-Enhanced Super Resolution GAN)**  
A GAN-based model designed for practical blind super-resolution of general image regions. Used here to enhance non-facial parts of each frame.

**🔑 Alpha Blending**  
A technique for merging multiple image layers with transparency. Used to combine the restored face region with the upscaled background seamlessly.

**D. Temporal Consistency**

**🔑 RAFT (Recurrent All-Pairs Field Transforms)**  
A deep learning model for dense optical flow estimation. It computes pixel-wise motion vectors between consecutive frames to track motion.

**🔑 Optical Flow**  
A representation of pixel motion between video frames. Essential for aligning frames temporally and ensuring smooth transitions.

**🔑 Frame Warping**  
The process of shifting an image based on optical flow. Used here to check whether the motion aligns well between frames.

**🔑 Temporal Loss**  
A loss function (often Mean Squared Error) that measures how well warped frames match actual frames, enforcing temporal smoothness.

**E. Output Generation**

**🔑 Re-encoding**  
The process of assembling individual enhanced frames back into an animated GIF or video format.

**🔑 ImageIO**  
A Python library used for reading and writing image data. In this project, it helps save the final enhanced GIF.

**F. Quality Evaluation**

**🔑 PSNR (Peak Signal-to-Noise Ratio)**  
A metric measuring the ratio between the maximum possible pixel value and the noise in the image. Higher PSNR indicates better denoising and restoration.

**🔑 SSIM (Structural Similarity Index)**  
A metric comparing the structural similarity between two images. Higher SSIM indicates better preservation of structures and textures.

**🔑 LPIPS (Learned Perceptual Image Patch Similarity)**  
A perceptual similarity metric based on deep neural networks. It better reflects human judgment of image similarity than traditional pixel-based metrics.

**G. Supporting Tools**

**🔑 Graphviz**  
A graph visualization tool that generates high-quality diagrams (like your architecture) using .dot files or Python code.

**🔑 Google Colab**  
A free cloud-based Python notebook environment that runs code interactively, often used for developing, testing, and visualizing ML pipelines.