REAL-TIME IMAGE ANIMATION USING DEEP LEARNING

A project report submitted in partial fulfillment of the requirements for

the award of the degree of

Bachelor of Technology

by

D. Srikar	**:**	**2111CS020557**
D. Srinath	**:**	**2111CS020560**
G. Srinidhi	**:**	**2111CS020561**
K. Varshini	**:**	**2111CS020617**
T. Varshitha	**:**	**2111CS020624**

Under the guidance of

Siva Kumar

Assistant Professor

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
MALLA REDDY UNIVERSITY
(As per Telangana State Private Universities Act No.13 of 2020 and G.O.Ms.No.14, Higher Education (UE) Department)
HYDERABAD – 500043
TELANAGANA
INDIA

2023-24

MALLA REDDY UNIVERSITY

(As per Telangana State Private Universities Act No.13 of 2020 and

G.O.Ms.No.14, Higher Education (UE) Department)

HYDERABAD – 500043

TELANAGANA

Certificate

This is to certify that this is the bonafide record of the application development entitled," REAL-TIME IMAGE ANIMATION USING DEEP LEARNING" submitted by **D.Srikar(2111CS020557),D.Srinath(2111CS020560),G.Srinidhi(2111CS020561),K.Varshini(2111CS020617),T.Varshitha(2111CS020624)** of B.Tech III year II semester, Department of CSE (AI&ML) during the year 2023- 24.The results embodied in thereport have not been submitted to any other university or institute for the awardof any degree or diploma

INTERNAL GUIDE	**HEAD OF THE DEPARTMENT**
	Dr. Thayaba Khatoon
Assistant Professor	**Professor & HoD**

ACKNOWLEDGEMENT

We would like to express our gratitude to all those who extended their support and suggestions tocome up with this application. Special Thanks to our mentor Siva Kumar whose help and stimulating suggestions and encouragement helped us all time in the due course of project development.

We sincerely thank our HOD Dr. Thayyaba Khatoon for her constant support and motivation all the time. A special acknowledgement goes to a friend who enthused us from the back stage. Last but not the least our sincere appreciation goes to our family who has been tolerant understanding our moods, and extending timely support.

ABSTRACT

This project delves into deep learning-based image animation, employing conditional generative models like Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). Trained on datasets comprising image-sequence pairs, these models transform single input images into coherent and novel animations, simulating natural movements and transformations. An interactive image animation system is introduced, implemented in a Jupyter notebook environment using TensorFlow for deep learning capabilities. Leveraging OpenCV, FFmpeg, ImageIO, PIL, and scikit-image for image and video processing, the system incorporates IPython widgets for enhanced user interaction. The technology also plays a crucial role in live video streaming, providing dynamic visual content without the need for manual frame-by-frame animation. This project harnesses the power of deep learning to eliminate manual efforts, opening new possibilities for efficient and realistic content creation in diverse domains.

CONTENTS

1. INTRODUCTION

This cutting-edge project pioneers the realm of image animation through the utilization of advanced deep learning techniques, specifically focusing on conditional generative models such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). By training on meticulously curated datasets containing pairs of images and their corresponding sequences, these models are adept at transforming static images into dynamic and cohesive animations, imbuing them with lifelike movements and transformations.

Central to this project is the introduction of an interactive image animation system, ingeniously implemented within a Jupyter notebook environment using TensorFlow for its robust deep learning capabilities. This system seamlessly integrates a host of powerful image and video processing libraries including OpenCV, FFmpeg, ImageIO, PIL, and scikit-image. Furthermore, it enhances user engagement and control through the incorporation of IPython widgets, fostering a fluid and intuitive interaction with the animation creation process.

Beyond its role in static image animation, this technology demonstrates its versatility by facilitating live video streaming with dynamic visual content, all achieved without the arduous task of manual frame-by-frame animation. By harnessing the unparalleled capabilities of deep learning, this project revolutionizes content creation across diverse domains, effectively eliminating manual efforts and ushering in a new era of efficient and hyper-realistic visual storytelling.

1.1. PROBLEM STATEMENT

The project tackles the inefficiencies and constraints of traditional image and video animation methods by harnessing the power of deep learning. It seeks to automate the animation process, ensuring that single input images are transformed into cohesive and lifelike animations without manual intervention. By focusing on realism and fluidity, the system aims to produce animations that exhibit natural movements and transformations. Through an intuitive user interface implemented in a Jupyter notebook environment, users can interactively control and customize the animation process. Additionally, the project aims to optimize computational efficiency, enabling real-time or near real-time animation generation. Extending its capabilities to live video streaming, the technology promises dynamic visual content without the need for laborious frame-by-frame animation, revolutionizing content creation across various domains.

1.2 OBJECTIVES OF PROJECT

Automated Animation Generation: Develop algorithms and models capable of automatically transforming single input images into coherent and dynamic animations, reducing or eliminating the need for manual frame-by-frame animation.

Realism and Cohesion: Ensure that the generated animations exhibit natural movements, transformations, and visual coherence, enhancing their realism and appeal to viewers.

Interactive User Interface: Implement an intuitive and user-friendly interface enabling users to interactively control and customize the animation process according to their preferences and requirements.

Computational Efficiency: Optimize algorithms and techniques to be computationally efficient and scalable, enabling real-time or near real-time animation generation even with large datasets and complex models.

Application to Live Video Streaming: Extend the capabilities of the system to support live video streaming, enabling the creation of dynamic visual content without the need for manual intervention, and thus opening up new possibilities for interactive and engaging multimedia experiences.

1.3LIMITATIONS

Training Data Quality: The quality and diversity of the training data may impact the realism and variety of generated animations.

Hardware Requirements: The computational resources required for training and inference may be substantial, potentially limiting accessibility to users with limited computing power.

Generalization: The models developed may exhibit limitations in generalizing to unseen data or scenarios, potentially leading to unrealistic or unexpected animation outputs.

Complexity of Animations: Generating complex animations with multiple objects or intricate movements may pose challenges for the models, potentially leading to less satisfactory results.

2. LITERATURE SURVEY

1. First-order Motion Model for Image Animation" by Aliaksandr Siarohin et al. (2019):

This paper introduces the first-order motion model, a deep learning-based approach for image animation. The model learns to transfer the motion from a driving video to a target image, generating realistic animations. The authors demonstrate the effectiveness of the model on various tasks, including face animation, object manipulation, and character animation.

2. Liquid Warping GAN: A Unified Framework for Human Motion Imitation, Appearance Transfer and Novel View Synthesis" by Mingyu Liang et al. (2019):

This paper presents the Liquid Warping GAN (LWGAN), a deep learning framework for human motion imitation, appearance transfer, and novel view synthesis. The LWGAN combines geometric warping with generative adversarial networks (GANs) to achieve high-quality image animation results across different domains, such as face animation and human motion imitation.

3. Few-Shot Adversarial Learning of Realistic Neural Talking Head Models" by Egor Zakharov et al. (2019):

In this paper, the authors propose a few-shot adversarial learning approach for generating realistic neural talking head models from a small number of input images. The method leverages deep learning techniques, including generative adversarial networks (GANs) and few-shot learning, to synthesize expressive talking head animations that closely resemble the input subject.

4. Deep Video Portraits by Justus Thies et al. (2019):

This paper introduces the concept of deep video portraits, where deep learning models are used to animate static portraits by transferring the motion from a source video. The authors demonstrate the capability of deep video portraits to generate high-quality animations of static images, including facial expressions and head movements, using a single input video.

5. Liquid Warping GAN++: A Unified Framework for Human Motion Imitation, Appearance Transfer and Novel View Synthesis with Improved Consistency and Quality" by Mingyu Liang et al. (2020):

Building upon their previous work, the authors propose an enhanced version of the Liquid Warping GAN (LWGAN++) framework for human motion imitation, appearance transfer, and novel view synthesis. The LWGAN++ improves consistency and quality in image animation tasks by incorporating additional loss functions and refinement mechanisms into the model architecture.

3. PROPOSED METHODOLOGY

3.1 EXISTING SYSTEM

The existing system for image animation typically relies on traditional computer graphics techniques and manual animation processes. Some common methods and technologies include:

Keyframe Animation: Animators manually define key poses or frames, and software interpolates between them to create smooth motion.

Motion Capture: Utilizing specialized hardware and software to capture real-world movements and apply them to digital characters or objects.

2D Animation Software: Tools like Adobe Animate (formerly Flash), Toon Boom Harmony, and others provide interfaces for creating 2D animations manually, often frame-by-frame.

3D Animation Software: Programs such as Autodesk Maya, Blender, and Cinema 4D allow animators to create complex 3D animations through modeling, rigging, and keyframe animation.

Motion Graphics Software: Applications like Adobe After Effects enable the creation of animated graphics and visual effects for videos, often using pre-built templates and effects.

While these existing systems offer powerful capabilities for animation creation, they often require significant manual effort and expertise. They may also lack the ability to automatically generate animations from single input images or offer real-time animation generation capabilities. As a result, there's a growing interest in exploring deep learning-based approaches to automate and enhance the animation process.

3.2 PROPOSED SYSTEM

The proposed system introduces a novel approach to image animation leveraging deep learning techniques, specifically conditional generative models like GANs and VAEs. The system aims to automate the animation process and enhance realism while providing an intuitive user interface for interactive control. Key components and features of the proposed system include:

- Deep Learning Models: Implementation of state-of-the-art conditional generative models trained on datasets containing image-sequence pairs to automatically generate animations from single input images.
- Realism and Cohesion: Focus on generating animations with natural movements and transformations, ensuring visual coherence and realism in the output.
- Interactive User Interface: Integration of an intuitive user interface within a Jupyter notebook environment, allowing users to interactively control and customize the animation process using IPython widgets.
- Efficiency and Scalability: Optimization of algorithms and techniques to ensure computational efficiency, enabling real-time or near real-time animation generation even with large datasets and complex models.
- Live Video Streaming Support: Extension of the system's capabilities to support live video streaming, enabling the creation of dynamic visual content without manual intervention.

By combining advanced deep learning techniques with an interactive user interface, the proposed system aims to streamline the animation creation process, enhance realism, and empower users to create compelling visual narratives with ease and efficiency.

3.3 MODULES

- Installation and Setup
- Widget Styling
- Widget Creation
- Functionality Implementation
- Animation Generation
- Output Options
- Widget Display

3.4 ARCHITECTURE

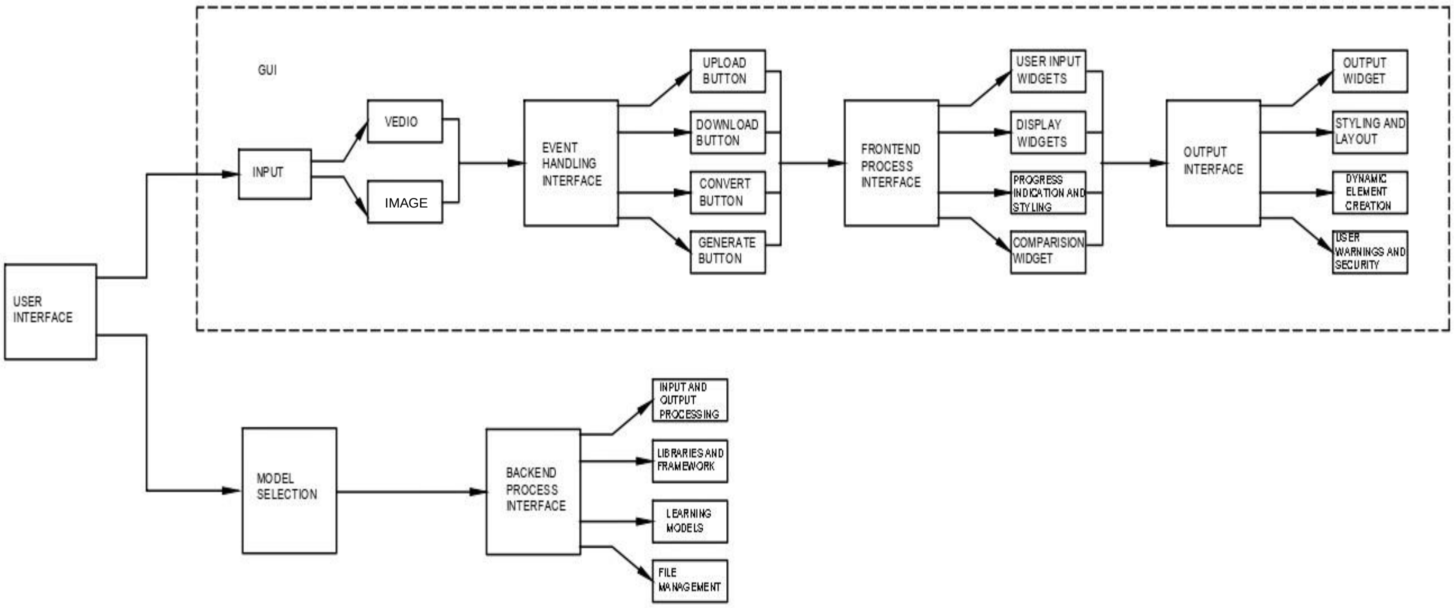

Fig no 3.4.1

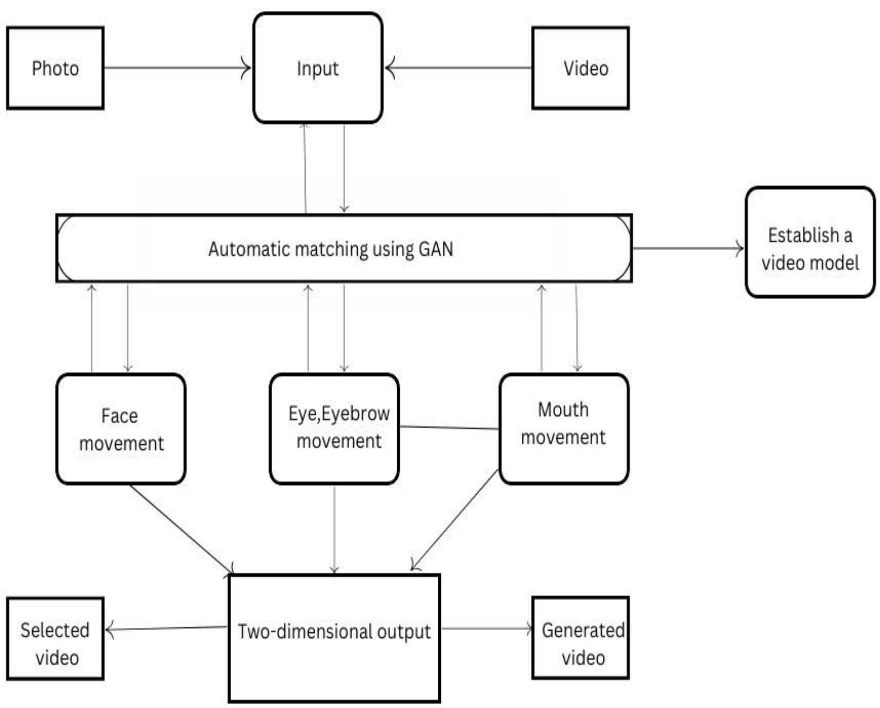

Fig no 3.4.2

3.5 METHODS AND ALGORITHMS

The primary deep learning algorithm used in the provided script is the First Order Model (FOM) for image animation. The FOM is based on conditional generative models and employs a combination of neural networks to generate realistic animations from single input images and driving videos.

First Order Model (FOM): The core algorithm used for image animation in the script is the First Order Model. FOM is a generative model that operates in a conditional manner, meaning it generates animations conditioned on both an input image and a driving video. It consists of two main components:

- Generator Network: This network generates the frames of the animated sequence based on the input image and the motion information extracted from the driving video.
- Keypoint Detector Network: This network detects keypoints or landmarks in both the input image and the driving video. These keypoints capture the spatial information necessary to align the generated frames with the motion dynamics of the video.

Conditional Generative Models: FOM belongs to the family of conditional generative models, where the generation process is conditioned on additional information (in this case, the input image and driving video). By conditioning the generation process, FOM can produce animations that are coherent and consistent with both the appearance and motion characteristics of the input data.

Neural Network Architectures: The generator and keypoint detector networks in FOM are typically based on deep neural network architectures. These architectures may include convolutional neural networks (CNNs) for feature extraction and transformation, recurrent neural networks (RNNs) for temporal modeling, and other components designed to capture complex dependencies between the input image and the driving video.