* **SharpVision AI – Project Report**

**Project Title: Image Sharpening using Knowledge Distillation**

**Team Details**

**Piyush -B.Tech AI&DS**

**Tanuj Yadav- B.Tech AI&ML**

**College: World college of Technology & Management , Gurgaon**

## ****Problem Statement:****

### Image Sharpening using Knowledge Distillation

* Many images captured by cameras in real life are **blurry or low in quality**.
* Clear images are important in fields like **security, medical imaging, and video calling**.
* Normally, we use **deep learning models** to improve image quality, but they are **heavy, slow**, and **use a lot of memory**.
* These big models can’t be used easily on **mobile phones or small devices**.

### Our solution:

* We used a method called **Knowledge Distillation**.
* A large model (called the **Teacher**) learns how to sharpen images well.
* A small model (called the **Student**) learns from both:
  + The real sharp image
  + The output of the Teacher model
* This helps the Student model become **lightweight, fast**, and still **very accurate**.

**Goal:**  
To build an **efficient and accurate image sharpening model** that works on devices with **low computing power**, without losing quality.

## ****Model Architecture****

Our project is based on **two neural networks** that work together: a **Teacher Model** and a **Student Model**. The idea is to train a smaller model (Student) to learn from a larger, high-performing model (Teacher), making the system faster and lightweight.

### ****1. Teacher Model****

* A **deep convolutional neural network** with many layers.
* Learns to convert blurry images into sharp images with high accuracy.
* This model is **large and powerful**, but too heavy for real-time or mobile use.
* Used **only during training** to guide the student model.

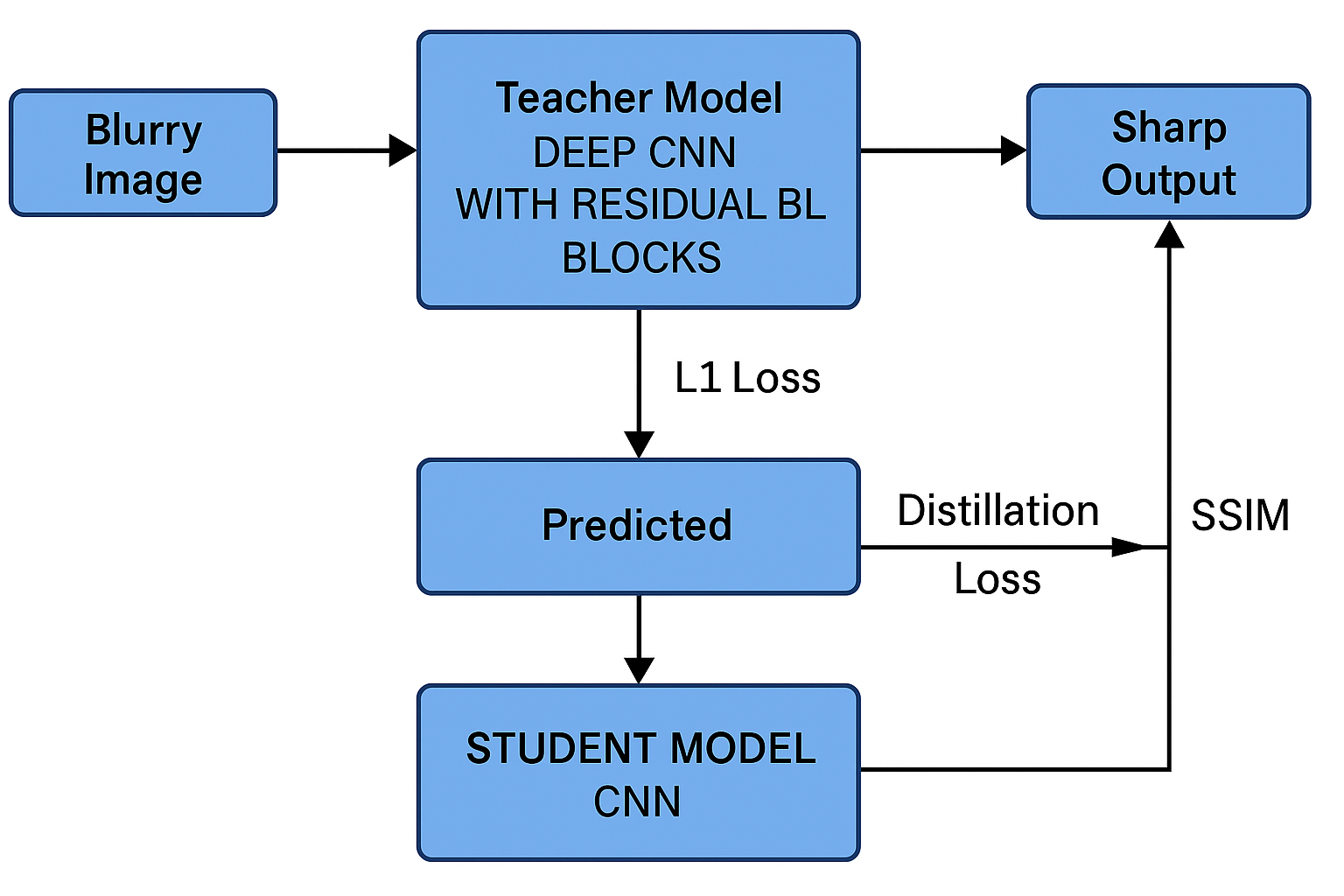
### ****2. Student Model****

* A **lightweight CNN** with fewer layers and much fewer parameters.
* Learns to produce sharp images by:
  + Comparing its output with the **ground truth** (real sharp image).
  + Also learning from the **output of the teacher model**.
* Can run on **mobile devices, IoT, and edge systems** after training.

### ****How the Training Works****

We used three key techniques during training:

* **L1 Loss** – Compares student output to the real sharp image.
* **Distillation Loss** – Helps the student learn from the teacher’s output.
* **SSIM (Structural Similarity Index)** – Used to evaluate image sharpness and quality.



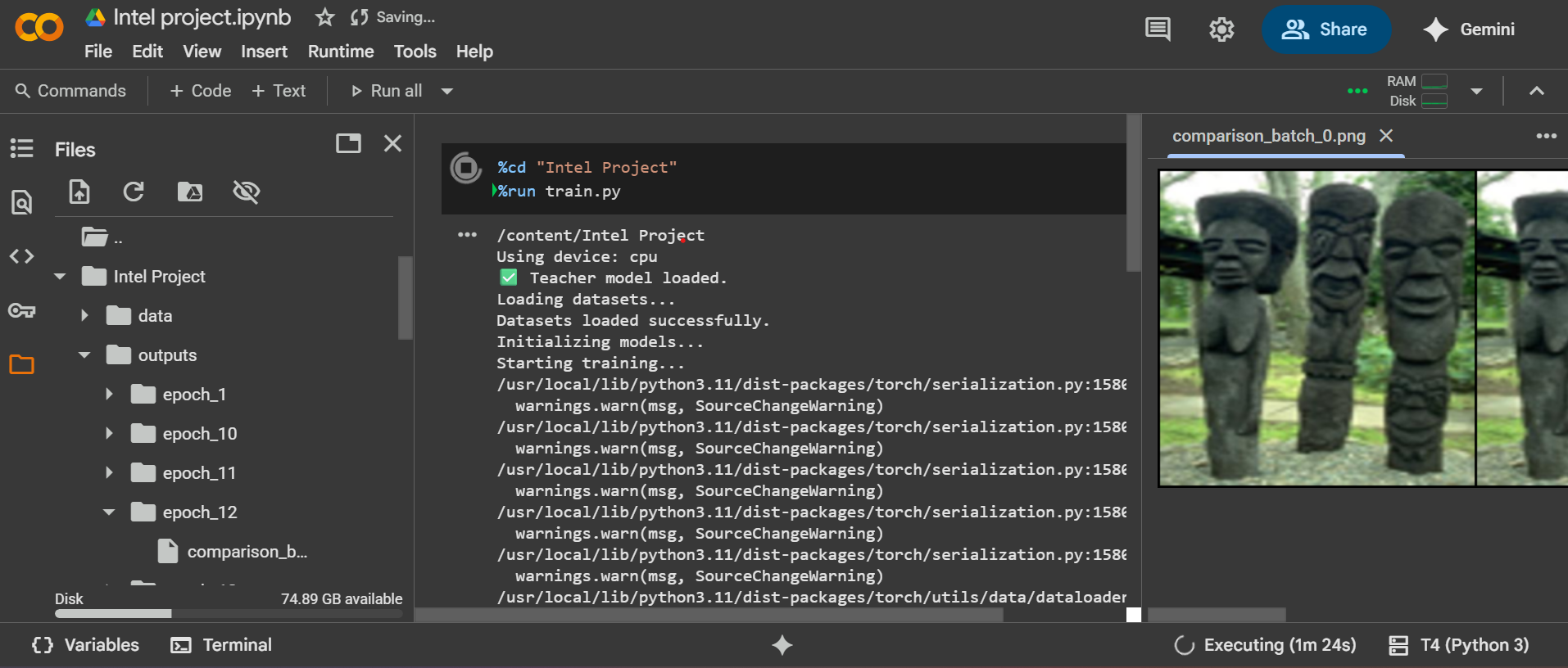
## ****Code Analysis****

* **models.py** – Defines Teacher and Student CNN models.
* **dataset.py** – Loads blurry-sharp image pairs for training.
* **train.py** – Trains the student model using L1 and distillation loss.
* **utils.py** – Calculates **SSIM is 0.7124** and handles image saving.
* **requirements.txt** – Lists needed Python libraries.
* **report.pdf** – Full documentation of the project.
* **results/** – Contains sample outputs (optional).

The code is clean, modular, and easy to run.

**Training Summary:** Project Explanation Video

* Dataset: Custom or open blurry-sharp image pairs
* Optimizer: Adam
* Epochs: 50+
* Evaluation: **SSIM is 0.7124**, PSNR comparison between student, teacher, and ground truth



**Project Highlights**

* Used **Knowledge Distillation** to build a lightweight model
* Achieved high image sharpness with minimal computation
* Created modular and scalable code with PyTorch
* Evaluated performance using **SSIM** and **visual comparisons**
* Open-source project hosted on **GitHub**
* Ready for edge deployment with reduced memory footprint

**GitHub Repository Link**

<https://github.com/Tanujydv123/SharpVision-AI-ImageSharpening>

**Project Explanation Video:**

[**https://drive.google.com/drive/folders/1HLDm\_MSMY\_H3YM1Oe\_BoOB-x5eZd7xYq?usp=drive\_link**](https://drive.google.com/drive/folders/1HLDm_MSMY_H3YM1Oe_BoOB-x5eZd7xYq?usp=drive_link)