**TECHNICAL REPORT**

**IMAGE SHARPENING USING KNOWLEDGE DISTILLATION**

**TEAM NAME : CODECRAFTERS**

**DATE OF SUBMISSION : 12/07/2025**

**ABSTRACT**

This project explores the innovative use of **Knowledge Distillation** to overcome the shortcomings of traditional image sharpening techniques. In many practical scenarios—especially on embedded or mobile devices—there is a pressing need for fast, reliable image enhancement tools that don’t demand heavy computational resources. Traditional filters, while simple, often lack adaptability and can even amplify noise instead of removing it. Our approach leverages the strength of a large, high-performing **Teacher model** and transfers its learned knowledge to a smaller, more efficient **Student model**. This allows us to maintain sharpness quality while minimizing the model size and inference time. Ultimately, this method provides a powerful solution that brings together the best of both worlds—deep learning accuracy and lightweight deployment.

**INTRODUCTION**

Image sharpening plays a critical role in a wide range of computer vision applications, from medical imaging and remote sensing to smartphone photography. The goal is to enhance an image’s fine details and edge clarity, making objects and textures more distinguishable. Traditional sharpening techniques such as unsharp masking or Laplacian filters are computationally light but fail to adapt to different image characteristics, often producing results that are either over-enhanced or excessively noisy.

In contrast, deep learning-based models are capable of learning intricate patterns and restoring finer details with high accuracy. However, they require significant computing power and memory, which limits their deployment on edge devices. Our solution addresses this trade-off using **Knowledge Distillation**, where a powerful Teacher model guides the training of a lightweight Student model. This approach achieves a balance between performance and efficiency—suitable for real-time and resource-constrained applications.

**MOTIVATION BEHIND THE PROJECT**

In today's digital world, the quality of visual data is more important than ever. From personal photography and social media to automated surveillance and smart healthcare, clear images are crucial. However, image quality is often compromised by motion blur, low-light conditions, or hardware limitations. Although traditional sharpening filters are fast and simple, they lack context-awareness and often make images look artificial.

Meanwhile, deep neural networks produce far better results, but they can be bulky and unsuitable for mobile or embedded platforms. This project was born from the desire to combine the **intelligence of deep learning** with the **practicality of lightweight models**. Using **Knowledge Distillation**, we extract the essential "knowledge" from a powerful model and distill it into a compact network, allowing us to sharpen images effectively—even in constrained environments. This innovation makes the technology more accessible, scalable, and real-world ready.

**DATA SOURCE**

For this project, we used the **DIV2K (DIVerse 2K Resolution) dataset** as the primary source for training both the Teacher and Student models.

The DIV2K dataset was introduced specifically for the **NTIRE challenge** (New Trends in Image Restoration and Enhancement), and has since become a gold standard in tasks where image fidelity is paramount. Every image in DIV2K has a resolution of at least 2K (minimum 1920x1080 pixels), ensuring that there are plenty of fine-grained details to train models effectively.

We selected DIV2K for several compelling reasons:

* **High-resolution and clarity:** The images are extremely sharp and well-defined, which provides a strong "ground truth" reference for the model to learn from. When a model is trained to replicate such detailed outputs, it gains a much better understanding of how a truly sharp image should look.
* **Wide diversity of scenes and textures:** The dataset spans a variety of settings—including nature, cities, architecture, close-up textures, and man-made objects. This diversity ensures that our models don’t just memorize patterns but learn to generalize across a range of visual content, which is crucial for real-world use.
* **Clean and uncompressed quality:** Unlike some datasets that contain compression artifacts or inconsistent quality, DIV2K images are clean and lossless, providing a reliable benchmark for training and evaluating image sharpening models.

To fully utilize the potential of the DIV2K dataset, we implemented a structured three-phase approach:

**1.Dataset Download and Preprocessing:**  
We used a Python script (<download_div2k.py>) to download and organize the dataset into training and validation folders. To simulate real-world scenarios, we applied controlled degradations such as Gaussian blur and downsampling using another script (<preprocess_images.py>). This process generated low-quality input images paired with their high-resolution originals.

**2.Teacher Model Training:**  
A deep convolutional neural network (such as UNet) was trained on these blurred–sharp image pairs using supervised learning. The Teacher model learned to recover details and sharpness effectively, though its complexity made it less suitable for deployment on lightweight platforms.

**3.Student Model Training via Knowledge Distillation:**  
Instead of training a smaller model independently, we used Knowledge Distillation. The Student model was guided by both the ground truth and the Teacher’s outputs and internal features. This dual-supervision approach helped the Student achieve similar performance to the Teacher while being faster and more resource-efficient—making it practical for real-time applications.

**WORK**

The development process followed a structured pipeline involving four main stages: teacher model training, student model design, knowledge distillation, and performance evaluation.

**1. Teacher Model Preparation**  
We began by training a deep CNN, such as **U-Net** or **ResNet**, on pairs of blurred and high-resolution images. This **Teacher model** learned to generate sharp outputs from degraded inputs through supervised learning. Its high capacity enabled it to restore fine details and serve as a benchmark for guiding the student model.

**2. Student Model Design**  
Next, we developed a smaller and more efficient **Student model** using architectures like **MobileNet** or a lightweight custom CNN. The Student was trained using a combination of:

* **Ground Truth Loss (L1/L2):** To match the ground truth image.
* **Feature Loss (Distillation Loss):** To mimic internal features of the Teacher.
* **Perceptual Loss (optional):** To enhance visual quality using high-level features from a model like **VGG**.

This allowed the Student to learn both output behavior and internal representation patterns from the Teacher.

**3. Training Pipeline**  
The Student model was trained using **Knowledge Distillation**, where it learned to replicate the Teacher’s predictions and features on the same blurred input images. This approach transferred valuable knowledge from the Teacher to the Student, helping it achieve comparable performance with fewer parameters and lower computational cost.

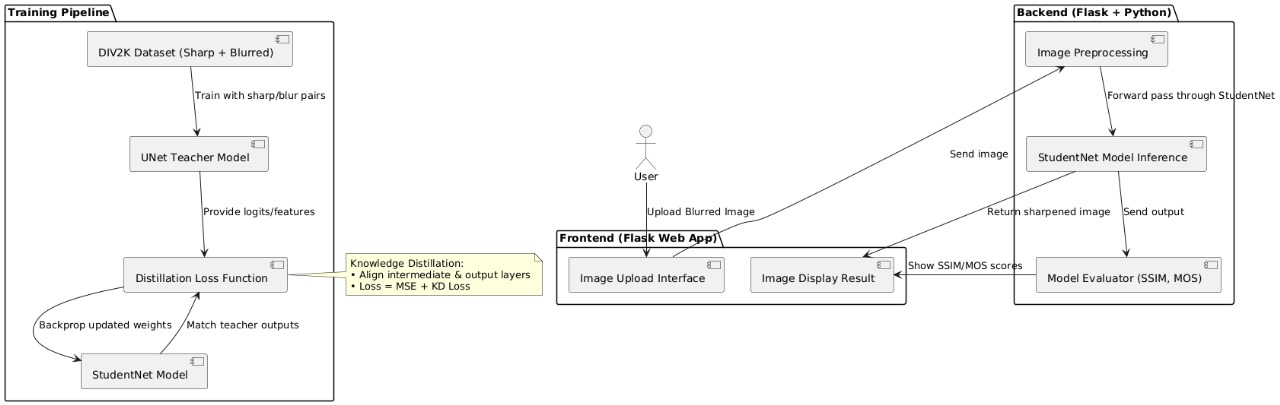
**4. Testing and Evaluation**  
We evaluated both models using key image quality metrics:

* **PSNR:** To measure reconstruction accuracy.
* **SSIM:** To assess structural similarity.
* **Laplacian Variance:** To quantify image sharpness.

In addition to metric-based evaluation, we performed **visual comparisons** between the input, Teacher output, and Student output to confirm that the Student produced visibly sharper results while being lightweight and efficient.

**SYSTEM ARCHITECTURE**

The system uploads images and displays the results using a Flask-based frontend. A lightweight Student CNN, trained through knowledge distillation from a deeper UNet Teacher model (used only during training), is used in the backend to process the image. Before being sent back to the user, images undergo preprocessing, postprocessing, and the student model. The DIV2K dataset is used for training, and SSIM and MOS are used to assess quality.



**System Architecture of Image Sharpening using Knowledge Distillation**

**RESULT**

A screenshot of a computer

AI-generated content may be incorrect.

**Fig 1: Image Sharpening Tool Interface**

Using the "Choose Image" button, users can upload a blurry image to improve its sharpness on a clean web page displayed by the interface.

A screenshot of a computer

AI-generated content may be incorrect.

**Fig 2: Image Sharpening Tool - Processing Stage**

The Model is improving the image's sharpness, as indicated by the interface's display of a blurry image being uploaded with a "Processing..." status.

A screenshot of a photo sharing tool

AI-generated content may be incorrect.

**Fig 3: Image Sharpening Tool - Before and After Comparison**

To illustrate the improvement, the interface shows a side-by-side comparison of the original blurry image on the left and the sharpened output on the right.

**LINK OF THE RESULT:**

**Github link :** <https://github.com/HarshithaReddy3339/ImageSharpeningUsingKnowledgeDistillation>

**TEAM MEMBERS AND CONTRIBUTION:**

* **Gullapalli Jayalakshmi,** worked on combining the student and teacher models using knowledge distillation, developed the image sharpening feature using neural networks, and handled the deployment of the project using Flask.
* **Gaddam Harshitha Reddy,** created the frontend interface, connected it with the backend, took care of preparing the images for processing, and helped implement the logic for sharpening images using the trained models.
* **Gundam Adarsh Reddy,** designed the overall look and layout of the user interface, prepared the project documentation, and helped in training the models and checking the quality of the results.