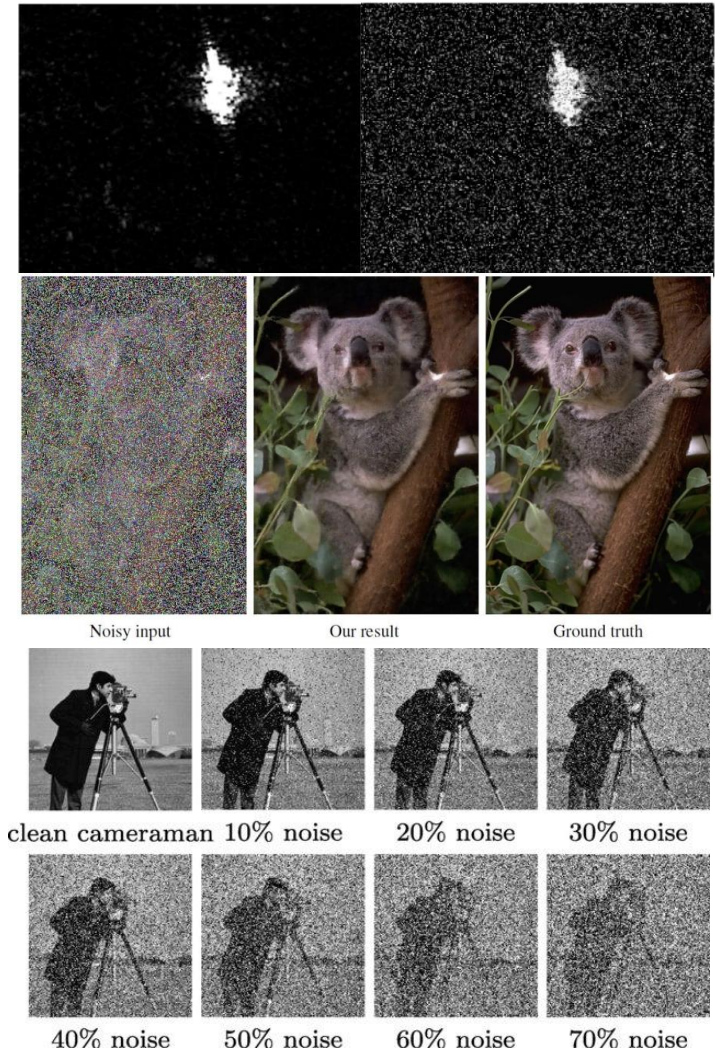


# Image Denoising with an Autoencoder

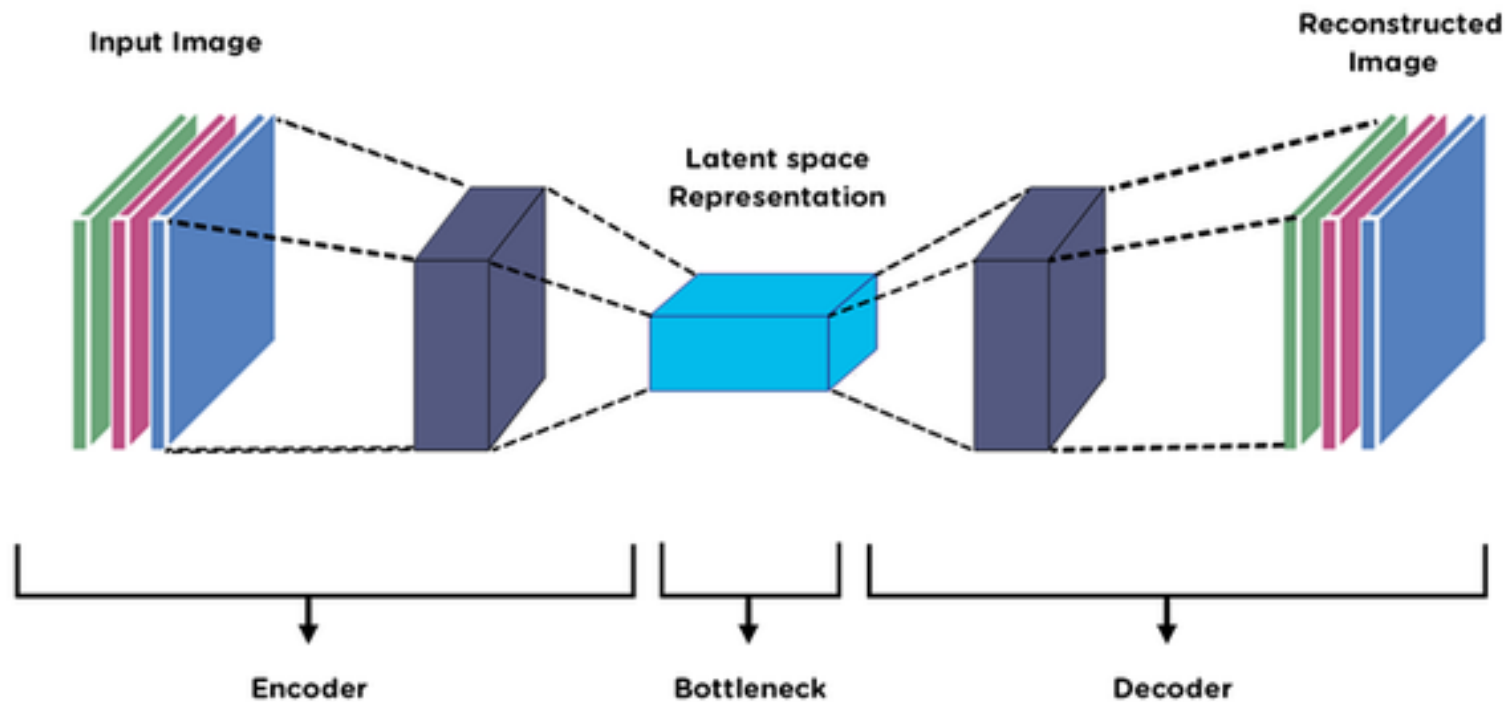
Aditya Hegde | AI and ML Club, SJSU

# Problem Description

- What is Image Denoising? – Essentially, recovering clean images from noisy input images.
- Why? – Noise severely degrades visual quality and reduces effectiveness of downstream tasks in other applications or domains.
- Image sources:
  - [MDPI](#)
  - [BleepingComputer](#)
  - [ResearchGate](#)



# Great, but what is an Autoencoder?



# What else can an Autoencoder be used for?

- **Image Colorization:** Converting grayscale images into realistic color versions.
- **Data Compression:** Reducing image dimensionality while retaining key features (useful for storage and transmission).
- **Super-Resolution:** Enhancing image resolution by reconstructing fine details.
- **Feature Extraction:** Learning compact latent representations for use in classification or clustering.
- **Anomaly Detection:** Identifying unusual patterns in images (e.g., defects in manufacturing, medical anomalies).
- **Style Transfers:** Mapping images from one style or domain to another (e.g., sketches → photos, conversions between different art styles).

# Domains or Applications

- Photography enhancement
- Satellite imagery
- Astronomy
- Medical Imaging
- Surveillance & autonomous systems

Just about any field dealing with imagery and suffering from noisy images is a candidate that benefits from Autoencoder architectures.

# Goals

Establish	- Establish baseline performance with a simpler autoencoder.
Explore	- Explore improved architectures (U-Net, residual autoencoders)
Experiment	- Experiment with real-world noisy datasets
Evaluate	- Evaluate with PSNR and visual inspection
Demonstrate	- Demonstrate downstream benefits for other CV tasks

# Data I've found thus far



[Low-Light Dataset](#): Approx. 40k images across ISO levels



[Darmstadt Noise Dataset](#) (DND): 50 high-res realistic noisy images, 12 GB in size



[Smartphone Image Denoising Dataset](#) (SIDDD): 30k images total

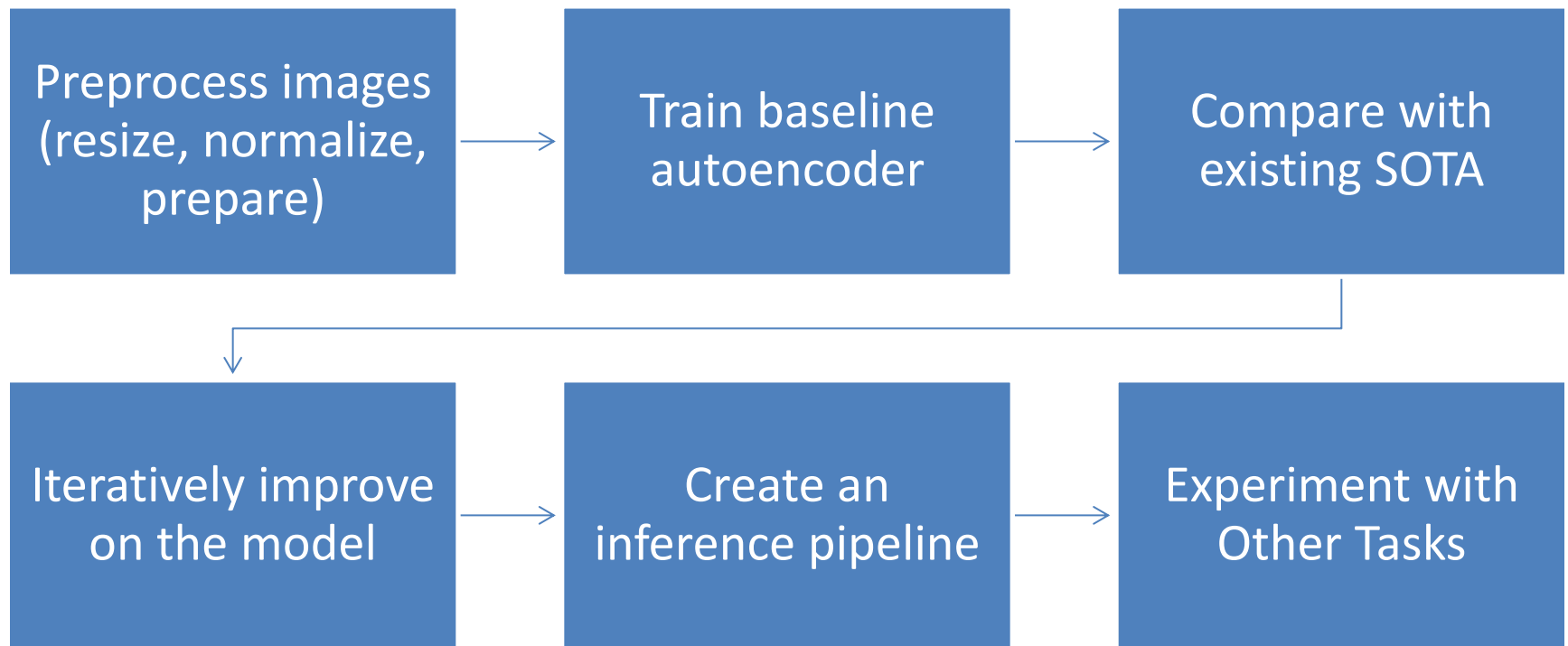


Synthetic noise augmentation (Gaussian, Poisson, salt-and-pepper)



Approach: start small → scale to larger realistic datasets

# Where do we go from here?





# Members

- I'm looking for 2-3 members
- Skills preferred but not mandatory:
  - Deep Learning Experience
  - Willingness to work with Pytorch
  - Git
  - OpenCV for image processing
- For now, all the code will be in python



# Interested?

Reach out to me on Discord (@aditya\_\_hegde) or just talk to me during the networking section of the meeting :)

P.S. the username has 2 underscores, not 1