Week 8 Assignment: Capstone Project Part 1

Candidate: Sneha Santha Prabakar

Coding platform: Google Collab

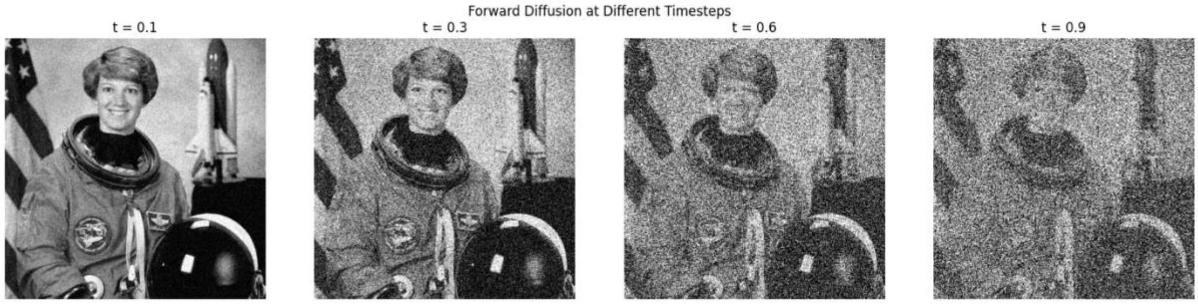
Part 1: Input sample image

Input Grayscale Image

This grayscale image is derived from a colour image using the rgb2gray() function, which applies a weighted sum over the RGB channels to retain perceived luminance.

- Grayscale conversion is essential in forward diffusion tasks to reduce dimensional complexity while preserving structural features.
- Pixel intensities are normalized between [0, 1] using img_as_float(), which facilitates consistent noise scaling and comparison.
- This step provides the clean baseline from which noise diffusion will be simulated in later stages.

Part 2: Forward diffusion at different time stamps

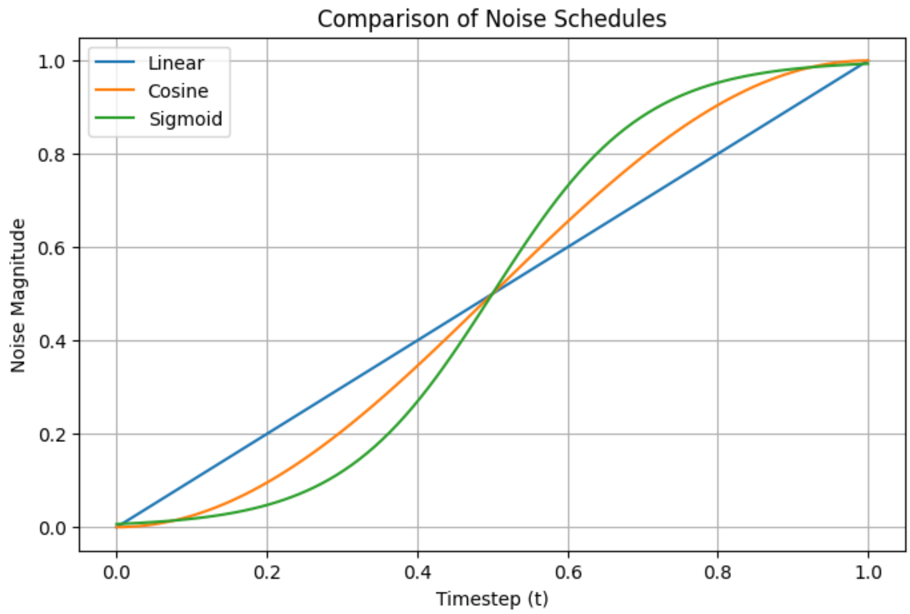


Each subplot shows the original grayscale image after Gaussian noise has been added at different noise levels (t = 0.1, 0.3, 0.6, 0.9).

Timestep t in diffusion represents the level of noise added - it linearly scales the standard deviation of the Gaussian distribution used to perturb the image.

- As t increases, the noise overwhelms the image content:
- At = 0.1, the structure and identity of the subject are still clearly visible.
- By t = 0.9, most fine details are obscured, showing how forward diffusion degrades information progressively.
- This mimics how diffusion models degrade data in the forward process before learning to reverse it.

Part 3: Apply noise schedule

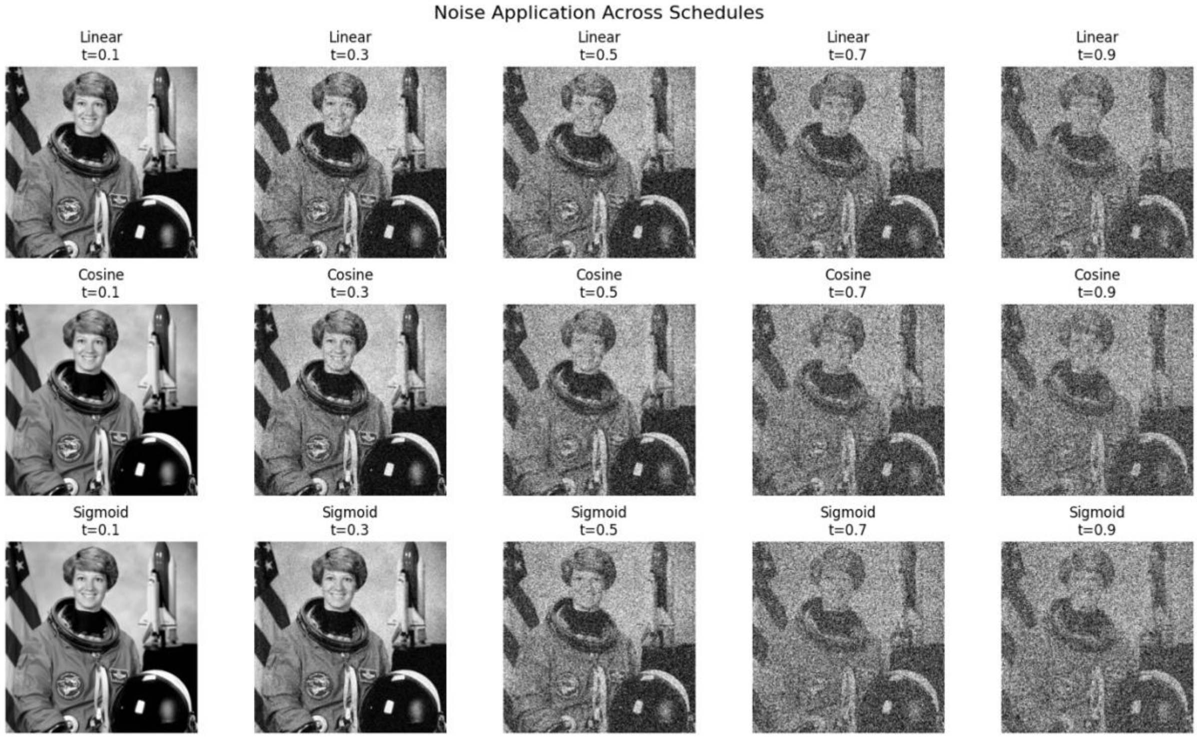


This plot shows how different noise schedules map timesteps (t) to noise magnitudes.

- Linear Schedule:
    - Adds noise at a constant rate across the diffusion process.
    - Easy to implement, but not always optimal for capturing complex image structures.
- Cosine Schedule:
    - Starts slow, then accelerates - better at preserving early image features.
    - Common in improved DDPMs due to its smooth growth and minimal early corruption.
- Sigmoid Schedule:
    - S-shaped curve with slow start and rapid middle transition.
    - Provides a balance between early detail preservation and late high noise.

Each schedule influences the learning dynamics of the model and how well it can reconstruct data from noisy samples.

Part 4: Noise application across schedules



Linear schedule:

Noise increases uniformly - details disappear steadily. Effective but can blur fine structure early on.

At **t = 0.1**:

- Very light Gaussian noise is visible.
- The subject's face and suit are clearly distinguishable.
- Minor speckling, mostly in uniform background regions.

At **t = 0.3**:

- Noticeable increase in noise.
- Texture begins to blur slightly.
- Edges and outlines are still intact, but image sharpness reduces.

At **t = 0.5**:

- Moderate noise level.
- Facial features start to lose sharp definition.
- The American flag in the background is less distinct.

At **t = 0.7**:

- Significant image degradation.

- Key features such as eyes and suit logos are hard to identify.
- Background nearly blends with the subject.

At **t = 0.9**:

- Heavy corruption of the entire image.
- Only the major silhouette of the person remains visible.
- Most texture is replaced by dense noise.

Cosine schedule

Cosine starts off gently and maintains structure in early timesteps, gradually ramping up noise. This makes it well-suited for preserving image features longer in the forward process.

At **t = 0.1**:

- Barely any visible noise.
- The image appears almost identical to the clean version.
- Excellent preservation of details.

At **t = 0.3**:

- A small amount of grain is introduced.
- Edges remain sharp and all features are still clearly visible.

At **t = 0.5**:

- Balanced level of noise is present.
- Slight blur appears, but facial structure and objects are distinguishable.
- Cleaner than linear at the same timestep.

At **t = 0.7**:

- Sharp rise in noise intensity.
- Details start to dissolve but are still somewhat visible.

At **t = 0.9**:

- High level of noise, similar to linear at this point.
- Silhouettes persist, but fine-grained detail is mostly lost.

Sigmoid schedule

Sigmoid applies very little noise in the early stages, then introduces a rapid burst of noise after the midpoint. Most dramatic noise jump appears between t=0.5 and t=0.7. This behaviour is ideal for preserving early image structure, while challenging the model with strong noise in the later stages.

- At **t = 0.1**:
  - No visible noise — nearly identical to the clean image.
  - Facial features and background remain pristine.
- At **t = 0.3**:
  - Still very low noise.
  - Subject is clear and well-defined with minimal corruption.
- At **t = 0.5**:
  - Abrupt jump in noise level.
  - Noticeably noisier than cosine or linear at the same point.
  - Detail starts to degrade suddenly.
- At **t = 0.7**:
  - Severe image degradation.
  - Loss of structure is much faster compared to earlier steps.
  - Image is much noisier than cosine at the same step.
- At **t = 0.9**:
  - The image is close to being completely corrupted.
  - Subject's identity is nearly impossible to discern.

Part 5: Compare diffusion models with GANs or VAEs, with a summary about noise and image structure

- Diffusion models offer a probabilistic framework where images are generated by reversing a gradual noising process.
- Unlike GANs, which learn a direct mapping from latent noise to data and often suffer from mode collapse, diffusion models explore the data manifold step-by-step, maintaining diversity and fidelity.
- VAEs encode data into a latent space and decode back, but often produce blurrier outputs due to the Gaussian prior assumption. In contrast, diffusion models capture fine-grained image structures through a learned denoising process. Their step-wise noise handling enables better control over image realism and detail restoration, making them highly effective for generative tasks, albeit with higher computational cost.