Step 1: Image Conversion and Grayscale

The image conversion begins by resizing color images and transforming them into grayscale using OpenCV's cv2.cvtColor. This step simplifies the image by reducing it to varying shades of gray.The resizing of the images ensures that the final ASCII output maintains a consistent and readable format. To account for the aspect ratio difference between images and ASCII characters, I used a width scaling factor to ensure that the final art would fit within the Gradio interface properly.

Step 2: Contrast Enhancement

In the next step, I applied Contrast Limited Adaptive Histogram Equalization[1] to the grayscale images. This enhancement brought out key features in the image that might otherwise be lost in the conversion to ASCII art. Using a clip limit of 2.0, the local contrast was improved, allowing for better differentiation between light and dark areas.

Step 3: ASCII Character Mapping and Edge Detection

To map pixel intensities to ASCII characters, I created a function that scales the pixel values to a corresponding set of characters: ascii_chars = ['M', 'N', 'B', 'F', '@', '#', 'S', '&', '%', '$', '*', '!', ';', ':', ',', '.', ' '] . Each character represents a different intensity level, with darker characters for high-intensity pixels and lighter characters for low-intensity pixels. I also included an optional edge detection feature using the Canny edge detection algorithm, which can further emphasize the outlines of objects within the image.

Experiment 1 (pass): I initially experimented with using double characters (e.g., 'MM', 'NN', 'BB') to increase visual density, but this approach led to clutter and overlapping, which reduced the clarity and detail of the resulting ASCII art.

Experiment 2 (pass): I then tried mapping pixel intensities to specific ASCII symbols using conditional thresholds (e.g., if pixel_value < 25: return 'M'). This method assigned a character to each intensity range but often resulted in uneven transitions and overlapping density between character selections.

To address this, I shifted to a more efficient approach by using a predefined list of ASCII characters and scaling the pixel intensity directly into that range. This allowed for smoother transitions between characters and avoided the visual clutter caused by manual thresholds.

Performance Metric:

1. SSIM (Structural Similarity Index):It provides an objective measure of how well the ASCII art retains the structural features of the original image. It calculates the similarity between the original and ASCII-converted images based on luminance, contrast, and structure, producing a score between -1 and 1. Higher SSIM scores indicate greater similarity. For instance, simpler images like a portrait might achieve an SSIM score of 0.75, while more complex images like a building could score around 0.60 due to the difficulty in representing intricate details in ASCII format.

[1] CLAHE is a technique used to improve contrast in images. It differs from ordinary histogram equalization in the respect that the adaptive method computes several histograms

2. Visual Evaluation: Assesses the human-perceived quality of the ASCII art in terms of clarity, balance, and readability.A visual score from 1 to 10 can be assigned based on clarity, balance, and readability.

Together, these two metrics provide a balanced assessment of both technical and perceptual quality.

Final Score = (0.5 * SSIM) + (0.5 * Visual Evaluation Score)

temporary 3-day web link for Gradio app: https://c73990313bd1765a3b.gradio.live

Permanent link using hugging face:

https://huggingface.co/spaces/Thisissophia/ascii-art-generator

Github: https://github.com/thisissophiawang/ascii-art-generator.git

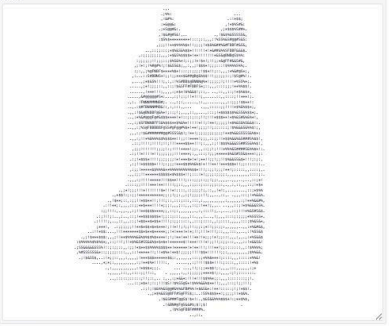
Image to ASCII Art Converter

Upload an image and convert it into ASCII art. You can also download the result as a .txt file.

Enable Edge Detection

Convert to ASCII

ASCII Art



Enable Edge Detection

Convert to ASCII

ASCII Art

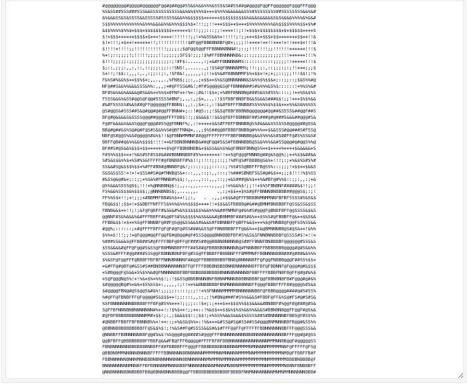
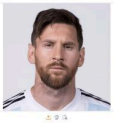


Image to ASCII Art Converter

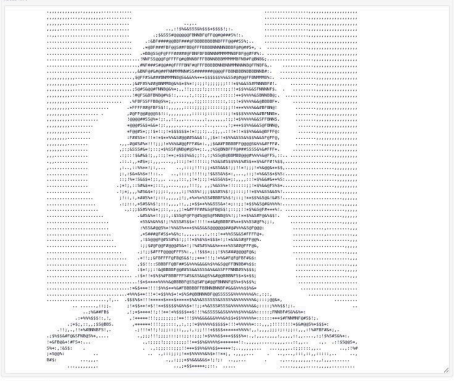
Upload an image and convert it into ASCII art. You can also download the result as a .txt file.



Enable Edge Detection

Convert to ASCII

ASCII Art



Enable Edge Detection

Convert to ASCII

ASCII Art