Photo-realism project pipeline:

1. Preprocess the single datasets
2. Apply **Histogram Matching** to the synthetic GTA images before the retrieval process. It Adjusts the color distribution of synthetic images to match the real images, which can make them look more realistic. This helps in better feature extraction and matching since the color distributions will be closer to those of the real images.
3. Use a **Retrieval algorithm** to create a gta-to-cityscapes dataset with the closest images where the real one is taken from cityscapes and the synthetic one from gta.
4. We load the GTA and Cityscapes
5. We use a pretrained Resnet18 (more accurate and lighter than vgg16) to extract features and cosine similarity to select the best pairs
6. Preprocess the gta-to-cityscapes dataset
7. Apply **Bilateral Filtering** on the gta images of gta-to-cityscapes dataset to reduce noise and preserving the edges. It reduces artificial noise while preserving important details. This way, the synthetic images can better match the natural quality of the real Cityscapes images, which remain unaltered to ensure that they provide a true-to-life reference for the model. (it may or may not help)
8. Train a pretrained **Diffusion model** (like Stable diffusion) on the gta-to-cityscapes dataset.
9. Divide the paired dataset into train and test set (mantaining the correspondence of similarity)
10. Apply random **Perspective transformation** to a random portion of the images in the training paired dataset to create variations. The training loader applies these transformations on-the-fly to each batch, providing diverse samples for each epoch. It creates dataset augmentation, increasing dataset variability, improving model robustness to various viewing angles and positions, ensuring diverse representations and enhancing model generalization.
11. Choose the model, train and test ( da fare)