"Machine learning for the comparison of synthetic images as a tool to support the evaluation of rendering exams"

Candidato: Anna Bondi

Relatore: Prof. Andrea Sanna

Prof. Federico Manuri

This master thesis takes place in the context of a rendering exam taken at Politecnico di Torino, where students , where students are given a Reference Image and have to reproduce it through rendering algorithms. Students are given a Blender file with scene and cameras already set and have to configure the correct settings in the 3D scene, such as lighting, textures, and materials. The aim of this thesis is to compare the Reference image given to the student and Render image produced by the student, in order to detect the changes occurred, focusing on specific features as lighting, position of the texture, transparency of the material, and reflections, using artificial intelligence. This tool provides a double benefit, the students will have assistance in the learning phase, understanding better their errors, and the professors will have a support toll for the evaluation

The task of finding differences in specific features on images takes the name of Semantic Change Detection, as it combine the Semantic Segmentation task, which is the identification of a specific object or feature in the image, with Change Detection, that is the identification of changes between two images. In literature, Semantic Change Detection is used in Remote Sensing, a field that has the aim to monitor changes that occur on earth's surface in the same area over time. Conceptually this project is similar, as it takes two images of the same scene, from the same viewpoint, and tries to detect changes in different features, such as illumination, reflections, textures and transparencies.

In this work, mainly two different neural networks for Semantic Change Detection were trained. The first, FresUNet, is based on U-Net architecture with skip connections, introducing residual blocks in each layer. The second, ChangeStar, uses a pre-trained feature extractor, which output is used to create change maps and segmentations, through a classifier and the ChangeMixin module. As feature extractor SAM (Segment Anything Model) from Meta was used, as it is a strong feature extractor trained on millions of images and billions of segmentation masks. In contrast, FresUNet is fully trained.
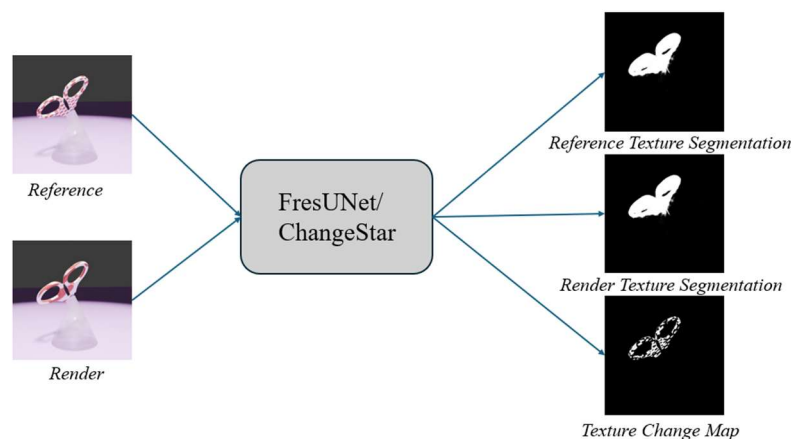

Figure 1: training pipeline

Both networks received as input the two RGB images (Reference and Render) and gave as output three binary images: two segmentations labels, one for the Reference and one for the Render image, and the change map relative to a specific feature [Figure 1].

To train these networks, four databases were generated with Blender, supported by the functionality this software offers, as Python API and render passes. Each feature was modified by manipulating a different parameter: i) illumination was changed with lights' parameters, ii) reflections with roughness of the material, iii) textures with the UV mapping, and iv) transparency with alpha value of the object. These parameters were dynamically changed between Reference and Render images, setting random values with a Python script, which also handled the rendering passes. In total five databases have been created. For each database, 2000 images where created, each one representing the variation of a single feature, except for one database that combined changes in illumination and textures. For each image, the label of the corresponding feature was generated through a combination of render pass, thresholding and transformations, to make the labels as detailed as possible. For each Reference image, 19 Render images were generated, and for each pair of Reference-Render images a Change Map of the feature to monitor was created. The Change Map is a binary image that indicates the pixels where the feature has changed. During the training, the annotations and the Change Maps were used as labels (Segmentation labels and Change maps respectively) to compute the loss. As both tasks of Semantic Segmentation and Change Detection are unbalanced problems, it was necessary to separately weight the losses, in order to have training performance as accurate as possible.

FresUNet and ChangeStar were trained independently with all databases, with the same hyperparameters and number of images, in order to have an equal comparison. Furthermore, the trained models were tested on two sets of images. The first set was composed of images in which only the feature trained was varying, the second set consisted of images where changes concerned all the features. Results were evaluated across visual evaluation and metrics like precision, recall, F1-score, and Intersection over Union. FresUNet consistently outperformed ChangeStar in terms of precision, F1-score, and the generation of accurate Change Maps, particularly in detecting shadows, textures, and transparencies. While ChangeStar exhibited higher recall in some cases, it often misclassified objects, leading to less accurate results. Both models struggled with complex scenes involving overlapping features and slight changes, highlighting the challenges in accurately detecting and segmenting fine-grained visual differences. Overall, FresUNet demonstrated greater reliability and precision, making it the more robust architecture for this task.