LCAV - Audiovisual Communications Laboratory

Semester projects 2020, Fall semester

Project description and planning

Project Description

This section is filled in by the project supervisor.

Title

Automatic Flat Colouring

Description (around half a page)

When creating the comic, the artists will create line-arts for every panel of the comic, that will have to be coloured. It's often more convenient to add so-called flat colours first and only later add shading and other effects. There are specialists, called flatters[1] whose job is exactly to prepare high-quality colour images. However, small projects and independent artists cannot afford to hire a flatter. This creates a need for automatic colouring of lineart.

There are existing solutions that provide automatic line-art colouring, but they all have their drawbacks. Images generated by neural networks look pretty and can deal with messy line-art[2] but they do not produce flat colours. Sometimes they change the line-art itself. It would be a desired behaviour if the output of the network were to be the finished picture, but in this case, we would like to have flat colours that will be the basis for shading.

The colouring G'MIC[3] filter produces good quality flat regions, but the colours are random. Tthere are no empty spaces below the lineart, the algorithm correctly guesses where the lineart should be connected even if there are gaps. G'MIC is based on a designed algorithm[4].

The goal of this project is to combine neural networks and G'MIC approaches to obtain an automatic colour selection and flat colours.

Type of Work (e.g., theory, programming)

Data processing (or generation) 30%, network architecture (or algorithm) design 30% and implementation 40%

Prerequisites (e.g., signal processing for communications, C++)

Experience in Python, basicsof machine learning and image processing or computer vision. Experience with Pytorch, TensorFlow, Kubernetes, ssh, C++, Pybind 11 or graph algorithms is a plus.

---

[1] https://en.wikipedia.org/wiki/Flatter

[2] Deepcolor: automatic coloring and shading of manga-style lineart

[3] GREYC's Magic for Image Computing

[4] S. Fourey, D. Tschumperlé and D. Revoy, “A fast and efficient semi-guided algorithm for flat coloring line-arts”, 2018.

Supervisors

Michalina Pacholska, Krzysztof Lis and Matthieu Simeoni

Student Information

This part is filled in by the student.

Name: Shuo Wen

E-mail: shuo.wen@epfl.ch

School (e.g., I&C, STI): I&C

Program (Comm. Sys., Comp. Science): Comp. Science

Cycle (B.Sc./M.Sc./EDIC): M.Sc.

Semester (1, 2, 3, ...): 3

Project Planning

*This part is filled in by the student with the help (and the agreement) of his supervisor. It should be completed before the end of the 2nd week (hard deadline on the **25th September 2020**) and sent to the responsible person (matthieu.simeoni@epfl.ch). After the submission of the plan, a modification is still possible, but it should be motivated at the midterm or the final presentation.*

Deliverables

Explain in a few sentences the expected concrete outcome of the project (e.g., a C program that removes red eyes, a Python simulation of sound propagation in a room, a subjective test on N persons of an algorithm).

Timeline

*Explain shortly (in a few sentences) what you plan to achieve for every week of the project in order to reach the final goal described in the previous section. Please remind that the students are supposed to spend **30 hours** to prepare the required background **before the beginning of the semester** (e.g., reading papers, revising Python/C). The amount of work during the semester should correspond to **18 hours** per week. After the end of the semester, **30 extra hours** should be spent to complete report and presentation, which gives a total of 312 hours of work.*

Week of	Planned work
14/09	Run pre-trained style transfer models and try them on different datasets.
21/09	Train style transfer models on the chosen dataset.
Project description and planning is due on Friday 25/09/20!	

28/09	Train style transfer models on the chosen dataset.
05/10	Run pre-trained pix2pix generation models and try them on different datasets.
12/10	Train pix2pix generation models on the chosen dataset.
19/10	Train pix2pix generation models on the chosen dataset.
26/10	Compare the two methods and analyze their pros and cons theoretically.
Midterm presentation	
02/11	Design the final model from the former analysis and train it.
09/11	Update the final model.
16/11	Update the final model.
23/11	Run GMIC from python.
30/11	Complete the pipeline.
07/12	Test our model on different dataset
14/12	Doing ablation study
21/12	Doing ablation study
08.01.2021	**Final report is due**
	Final presentation
25.01.2021	**Final Grade is due**