Prof. Dr. Thomas Schultz
Mohammad Khatami (khatami@cs.uni-bonn.de)
Ikram Jumakulyyev (ijumakulyyev@cs.uni-bonn.de)

University of Bonn
B-IT/Computer Science II
June 12, 2020

Summer term 2020

Visual Data Analysis

Assignment Sheet 8

Solution has to be uploaded by June 22, 2020, 8:00 a.m.
to `https://uni-bonn.sciebo.de/s/O5AqUl89e65E3CV` with password `vda.2020`

Please bundle the results (as PDF) and scripts (*.py/*.ipynb files) in a single ZIP file. Submit each solution only once, but include names and email addresses of all team members in the PDF and each script. Name the file `vda-2020-xx-names.zip`, where `xx` is the assignment sheet number, and `names` are your last names.

If you have questions concerning the exercises, please write to our mailing list:
vl-scivis@lists.iai.uni-bonn.de.

Exercise 1 (Introduction to ParaView, *13 Points*)

In this exercise we introduce you to ParaView, a cross-platform data analysis and visualization application that implements most algorithms we will discuss in the next few weeks. With ParaView you can quickly visualize spatial data from simulations or medicine. You can download the recent version of ParaView from `https://www.paraview.org/download/`. Please send us both a screenshot and the corresponding Paraview state (`*.pvsm` file) of your solution.

Hint: On Windows, an MPI and a non-MPI version of ParaView are available. The non-MPI version is sufficient for our purposes and should work "out of the box" after running the installer. Getting the MPI version to run requires installing additional libraries.

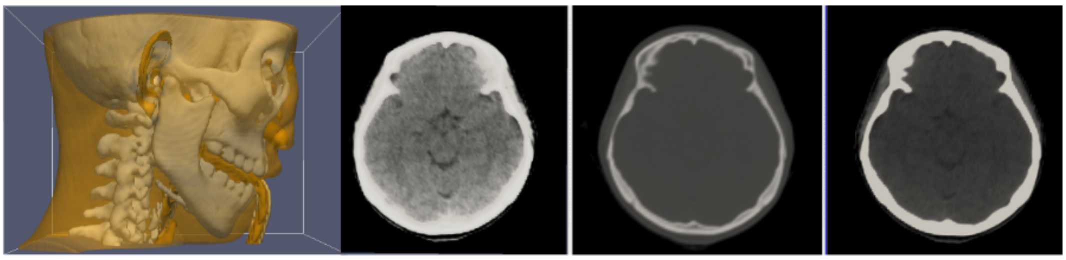

Figure 1: Nested 3D surfaces (left) and brain slices with different colormaps (right).

a) Load the file `headsq.vti` into a ParaView pipeline. What is the grid type of this dataset? What is the type and range of its values? (2P)

b) A basic and widely used way to visualize 3D CT data is via 2D slices. Different mappings from Hounsfield units to grayscale are used depending on the structures of interest. To examine brain, bone, or potential intracranial hemorrhage, common ranges ("windows") are $[0, 80]$, $[-770, 1730]$ and $[-85, 265]$ Hounsfield units, respectively. Create three corresponding images, which should look similar to the ones in Figure 1. (6P)

Hint: The values in the given input file correspond to shifted Hounsfield units: 1024 has been added in order to store them as unsigned values.

c) Also create a 3D visualization showing two nested surfaces as in the left part of Figure 1: The outer surface should show the skin, and it should be rendered in a different color as well as semi-transparently in order to reveal an inner surface, which should visualize the bones. (5P)
Hint: This can be achieved with isosurfaces, which are available via ParaView's `Contour` filter.

Exercise 2 (Visualization Pipeline in Python, *12+4 Bonus Points*)

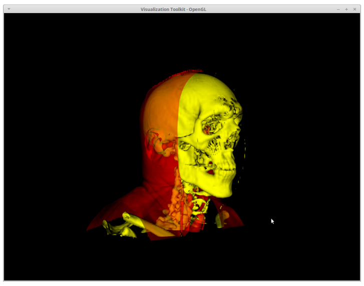

Figure 2: Final result of the Python script.

In this exercise, you will use VTK's Python bindings to implement a simple visualization pipeline for volumetric data. The goal is to read the file `head.vti` (available from the lecture homepage) and create a visualization similar to the one in Fig. 2. On the lecture homepage, we provide a framework that already implements some of the functions you will need. We have marked the areas where you have to add code yourself with "<Insert Code>".

a) Finish the function "task1": Create a bounding box for the input data using an outline contour filter (vtkOutlineFilter) and set the color of the bounding box to green.
b) Finish the function "task2": Create a contour for the bones by vtkContourFilter and set the color to yellow. You may have to try multiple isovalues. Which one is suitable for extracting the bone structure?
c) Finish the function "task3": Create a contour for the skin, set the color to red and make it transparent. Also add the clipping plane using vtkPlane.
d) In the main function, implement the part which disables the visibility of the bounding box when ShowBBox is not set, so that the bounding box is only shown when the '-b true' parameter is used at the command line.

Please submit a screenshot of your final result and your Python script.

Hint: You can find various python-VTK example at `https://lorensen.github.io/VTKExamples/site/Python/` . Looking at some of the examples may help you to create and properly connect VTK modules in Python. When working with the VTK classes, you will frequently have to refer to the documentation at `https://vtk.org/doc/release/7.1/html/`.

Bonus Task: Add another command line parameter "-a true" that enables an animation in which the camera is rotated around the dataset.

Hint: In order to do the animation, you will have to create a function that will be executed automatically after a given amount of time (i.e., triggered by a timer). In this function, you will have to get the active camera that is used in the renderer and rotate it. You will need to call several Get<SOMETHING>() functions starting from the interactor.

Exercise 3 (Natural Pre-Images for Visualizing CNNs, *11 Points*)

In the lecture, we discussed different strategies to generate images that visualize what a specific neuron in a convolutional neural network responds to, or what is represented in a given layer. This is also the topic of `mahendran-natural-pre-images-2016.pdf`, which is available from the lecture webpage. Please answer the following questions about it in your own words. As always, we do not give credit for copy-pasted text.

a) Briefly explain the respective roles of the three terms in the objective function presented in the paper's Eq. (1). (3P)

b) What is the role of the constant C in Eq. (1)? What happens when it is set to very large or very small values? Based on the experimental results that are reported in the paper, name a factor that affects what is a sensible value of C. (3P)

c) The authors mention that, in Fig. 13, "the effective receptive field of the neurons is in some cases significantly smaller than the theoretical one". Highlight some specific subfigures in which that is the case. How can you tell? What could be the reason for this? (2P)

d) Which images that were shown in the lecture are most similar to Fig. 16 from this paper in terms of their intended interpretation? (1P)

e) Assume you are considering to use a cloud service that offers to apply a set of neat artistic filters to your personal photos. Their service is for free, but they require your permission to share the activations from the CNNs on which their filters are based with their advertising clients. They assure you that your images themselves will never be shared. Would you have any privacy concerns about using such a service? Briefly justify your answer. Would it change if they promised to only share activations from the second fully connected layer onwards? Why? (2P)

Exercise 4 (Grids, *8 Points*)

Please submit answers to the following questions:

a) How many numbers need to be stored to fully specify a scalar field on a Cartesian grid of size $10 \times 10 \times 10$ (including the data itself)? (2P)

b) How many cells does the grid from a) contain? (2P)

c) How many *additional* numbers need to be stored if the same grid is specified in more general form as a rectilinear grid? (2P)

d) Given a rectilinear grid, what criterion has to be checked to decide if it can also be represented as a uniform grid? (2P)

Exercise 5 (Bilinear and Barycentric Interpolation, *6 Points*)

You are given points A (0,0), B (0,1), C (1,1) and D (1,0) on a plane with the following values: A=1, B=2, C=3, D=4. Calculate the value of E=(1/4,1/2) by bilinear interpolation and barycentric interpolation with respect to the two possible triangulations in the 2nd and 3rd drawing in Figure 3.

Good Luck!

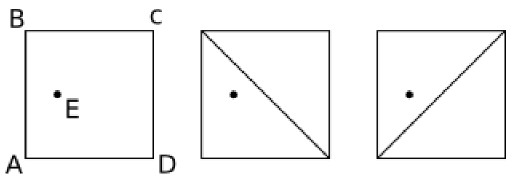

Figure 3: A grid cell ABCD and a point E at which to interpolate.