CS/ECE 4501 / CS 6501 / ECE 6782 **Due by Oct. 11** $11:59:59$**PM**

Problem Set II

1. (CS 6501 / ECE 6782 30%) (CS/ECE 4501 40%) **Euler's method:** `https://en.wikipedia.org/wiki/Euler_method`

   For $\frac{dy}{dt} + 2y = 2 - e^{-4t}, y(0) = 1$,

   (a) Derive its closed-form solution on your own.

   (b) Use Euler's Method to find the approximation to the solution at $t = \{1, 2, 3, 4, 5\}$, and compare to the exact solution in (a) by plotting them on a same figure.

   (c) Use different step size $h = \{0.1, 0.05, 0.01, 0.005, 0.001\}$ and plot out your approximated function value.

2. (CS 6501 / ECE 6782 50%) (CS/ECE 4501 60%) **Generating time-sequential images with geodesic shooting equations.** Given an initial velocity, implement geodesic shooting algorithm (via Euler integration) to generate a time-sequence of deformations (transformations), $\phi_t, t \in [0, \cdots, 1]$. You will then deform a given source image (included in the data folder) by using the final transformation ϕ_1 at time point $t = 1$.

   Note that the initial condition v_0 is given in the data folder, and the initial condition for ϕ_0 is an image coordinate of the given source image, which can be easily generated from Python (e.g., numpy's meshgrid).

   Your task will be:

   (a) Implement and compute the geodesic shooting equation below (a special case of the original shooting equation discussed in class) using Euler integration with 10 time steps, e.g., $t = \{0, 0.1, 0.2, \cdots, 1\}$.

   $$\frac{dv_t}{dt} = -K[(Dv_t)^T \cdot v_t + (Dv_t) \cdot v_t + v_t \cdot \mathrm{div}(v_t)],$$
   $$\frac{d\phi_t}{dt} = v_t \circ \phi_t,$$

   where K is a smoothing kernel (e.g., a Gaussian smoothing kernel), D is a Jacobian matrix, div is a divergence operator, and $\circ$ denotes an interpolation.

   (b) Report the final velocity $v_{t=1}$ and a time-sequence of deformed source image s by computing $s \circ \phi_t$.

   (c) Report the total running time of your shooting algorithm.

   (d) Now generate your own random initial velocity field by computing $\epsilon \cdot \nabla s$, where ϵ denotes randomly generated velocity fields drawn from a normal Gaussian distribution followed by being smoothed by a Gaussian kernel (try different smoothing variances with values of $2.0, 4.0, 8.0$). Repeat (a)-(b).

   * Interpolation function: Python function *scipy.ndimage.map_coordinates* with the option 'order=3'.

3. (**CS 6501 / ECE 6782 ONLY.** 20%) Compute the inverse transformation ϕ_1^{-1} at time point $t = 1$ by the following strategy.

$$\frac{dv_t}{dt} = K[(Dv_t)^T \cdot v_t + (Dv_t) \cdot v_t + v_t \cdot \mathrm{div}(v_t)],$$
$$\frac{d\phi_t^{-1}}{dt} = -D\phi_t^{-1} \cdot v_t.$$

Report the deformed source image using ϕ_1^{-1}, and describe how it is different from the final deformed image in problem 2.

IMPORTANT NOTES:

* All results should be clearly reported and discussed in the report.

* This is not required, but students in CS/ECE 4501 are welcome to use Q3 for bonus points.