Numerical Simulation of Mirages above Water Bodies

Balázs Bámer, Anna Horváth, Gergely Gábor Barnaföldi

June 21, 2022

The research was supported by the Hungarian National Research, Development and Innovation Office (NKFIH) under the contract numbers OTKA K123815 and 2020-2.1.1-ED-2021-00179. Computational resources were provided by the Wigner Scientific Computing Laboratory (WSCLAB) (the former Wigner GPU Laboratory).

Outline

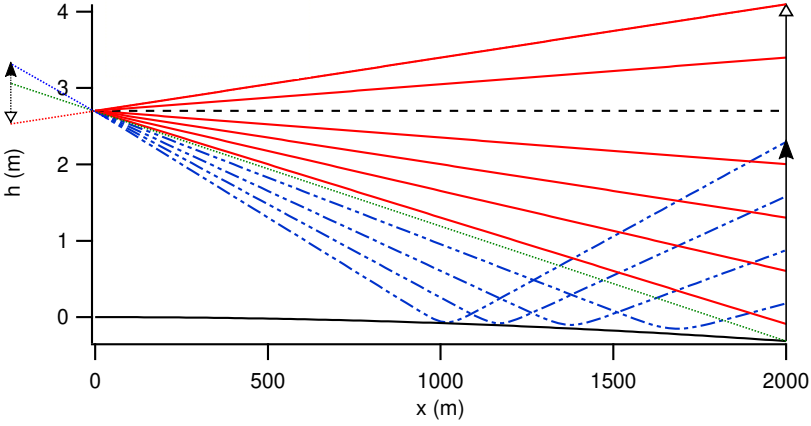
Motivation

Mirages can be seen very often in the nature as well as in urban environment. Simulating it is simple but interesting task, which enables building model and a further application to deduce air-water temperature difference from a photo.

- Inferior mirage is formed over a surface warmer than the ambient air.
- This phenomenon occurs over various surfaces like
  - water
  - asphalt
  - grass, sand etc

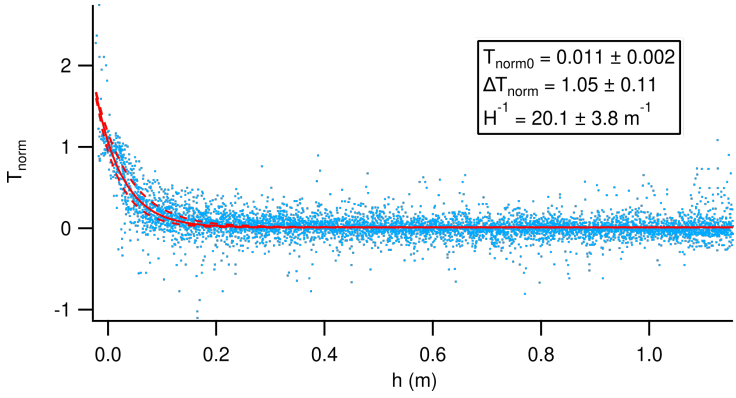
Refraction of light and mirages

Light bends towards the greater refraction index



- White arrow: full object
- Red lines: rays of normal image
- Black arrow: object part in mirage
- Blue lines: rays of mirage
- Green line: horizon
- Dotted black line is pinhole image

Temperature profile and model



Our model is

$$T(h) = T_a + \Delta T e^{-h/H},$$

$$n = 1 + \frac{0.786p}{T + 273.15}$$

Where p is pressure in Pa
T is temperature in Celsius

Eikonal equation

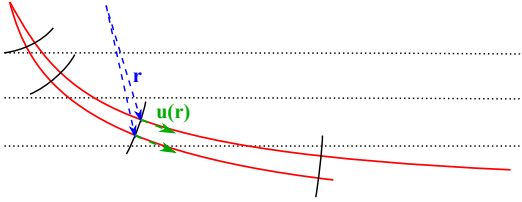
The eikonal equation is

$$|\nabla t^2| = \frac{1}{v^2},$$

which can be derived into the following system of equations:

$$\frac{\mathrm{d}\,\mathbf{r}}{\mathrm{d}\,s} = v(\mathbf{r})\mathbf{u}(\mathbf{r}),$$

$$\frac{\mathrm{d}\,\mathbf{u}}{\mathrm{d}\,s} = -\frac{1}{v(\mathbf{r})^2}\nabla v(\mathbf{r}).$$

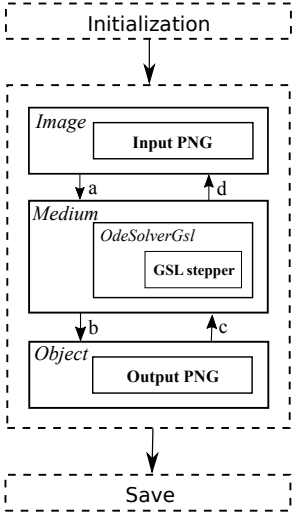


where

- t is wavefront travel time
- v is the actual speed of light
- s is the length along the ray
- r is the position vector
- u is the slowness vector defined by

$$\frac{1}{v(\mathbf{r})}\frac{\mathrm{d}\,\mathbf{r}}{\mathrm{d}\,s}$$

Structure of the program code



Properties:

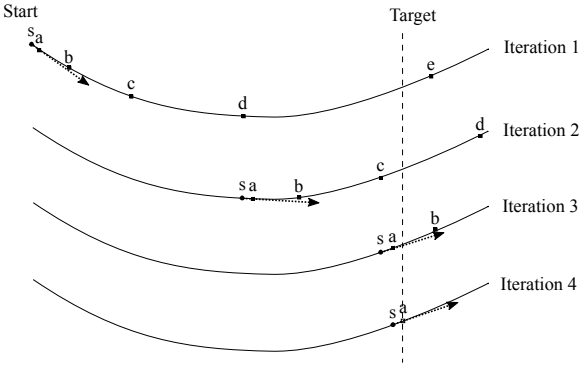
- PNG input/output
- Standard C++17
- Parallel execution on all cores
- Command-line app to allow batch execution

Used libraries:

- GNU Scientific Library
- Eigen3
- libPNG
- png++

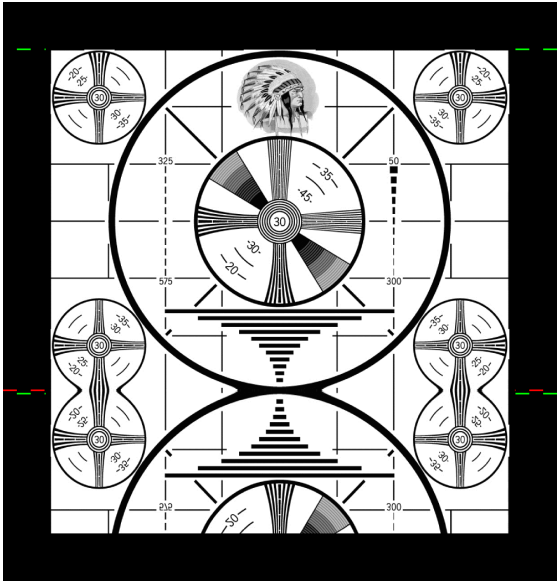
OdeSolverGsl is a general-purpose ODE system solver for arbitrary end condition.

The integration algorithm



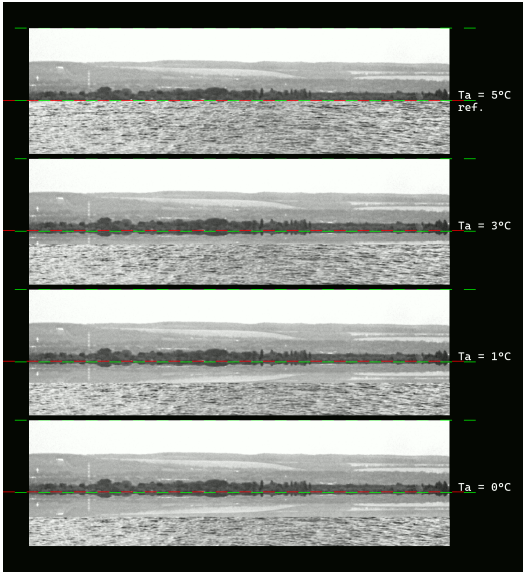
- Odeiv2 in GSL is for a system of ODEs.
- Here the integration goes until some geometry defined in the dependent variable.
- Adaptive Runge – Kutta – Fehlberg (4, 5) stepper algorithm
- Integration restarted from the last step before the geometry condition.
- Stops when the step exceeding the condition is small enough.
- General templated C++ solution for this problem.
- Computational cost is $\mathcal{O}(-\log x \log \epsilon)$

Simulation output

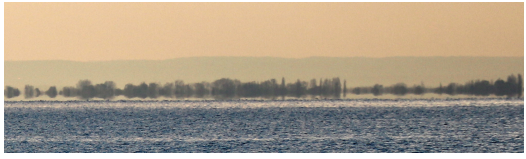


- Original RCA monoscope shows distortion quite well.
- $T_{amb} = 2°\text{C},\ T_{water} = 13°\text{C}$
- Green lines show the image limits for $T_{amb} = T_{water}$
- Red line is the horizon
- Bottom part of the object is invisible

Simulation of a real-life scenario



- Photos taken on lake Balaton, in Keszthely in direction to Balatonfenyves
- Distance of object is around 16.8 km.
- $T_{amb} = 0°\text{C},\ T_{water} = 5°\text{C}$
- Water is rendered artificially.



Summary and future work

Goals

- built a simple temperature model for mirage simulation
- developed a general C++ lib to solve ODEs with arbitrary end condition
- developed a simple ray tracer app to simulate mirages
- compared the simulation results to real-life mirage photo

Possible future work includes

- image processing using AI to deduce air-water temperature difference from photos
- simulate and observe mirages over asphalt
- implement the algorithm on GPU
- implement wavelength-dependent simulation for accurate investigation of real scenes

Animations:

https://indico.wigner.hu/event/1393/contributions/3157/attachments/2238/4367/chess.mp4

https://indico.wigner.hu/event/1393/contributions/3157/attachments/2238/4366/balaton.mp4

Code:

https://gitlab.wigner.hu/bamer.balazs/numerical-simulate-mirage.git

Thank you for your attention.