The Andrew & Erna Viterbi Faculty of Electrical Engineering Technion — Israel Institute of Technology

Electronics
Computers
Communications

VISL Vision and Image Sciences Laboratory

Final Presentation

Turbulence Simulator

Students: Simon Shvarts, Lidor Malul

Supervisor: Marina Alterman

Context: Project A

Semester: Spring 2018

Date: 20/02/19

In Collaboration with: RAFAEL

RAFAEL ADVANCED DEFENSE SYSTEMS LTD.

Presentation Outline

- Project goal
- Background
- Optional solution
- Chosen solution
- Results
- Summary
- Conclusions

Project Goal

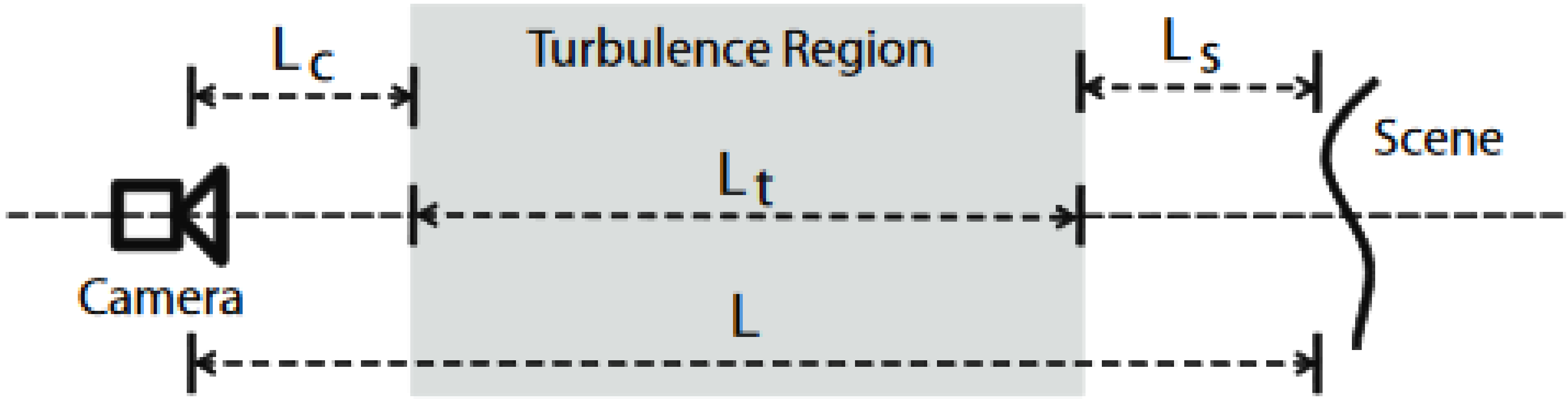
- Simulate distortion of an image caused by atmospheric turbulence using given parameters of user input.
- Create a video from simulated images.
- Create a data base of pictures affected by turbulence for machine learning algorithm.

Background

- A long-distance imaging system can be strongly affected by atmospheric turbulence.

- random changes of the refractive index along the optical transmission path.
- causes geometric distortion (motion), space and time-varying blur.

Physical Background



(b)

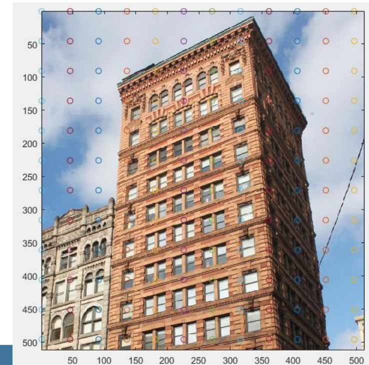
Physical Background

- Turbulence strength is determined by Cn
- Cn is function of temperature, pressure and more.
- $\langle \alpha^2 \rangle = 2.914\mathrm{D}^{-\frac{1}{3}} \int_0^L C_n^2(z) \cdot \left(\frac{z}{L}\right)^{\frac{5}{3}} dz$
- α is the arrival angle.
- D is the camera's aperture.

Optional Solutions

- Random pixel movements
- Milanfar algorithm – statictical and physical

Milanfar solution: Block Diagram



Choose m reference points → Create random movements vector in x and y directions → Calculate C1-Cm for each pixel in the picture

→ Create A(x) matrix → Calculate the total movement matrix → Generate the simulated picture using "int2" function in MATLAB

Choose m points in a picture

Milanfar solution: details

- Choose m points
- Draw movements vector p
- Calculate Ci for every pixel and calculate A(x)

$$\mathbf{A}(\mathbf{x}) = \begin{bmatrix} c_1 & \dots & c_m & 0 & \dots & 0 \\ 0 & \dots & 0 & c_1 & \dots & c_m \end{bmatrix}$$

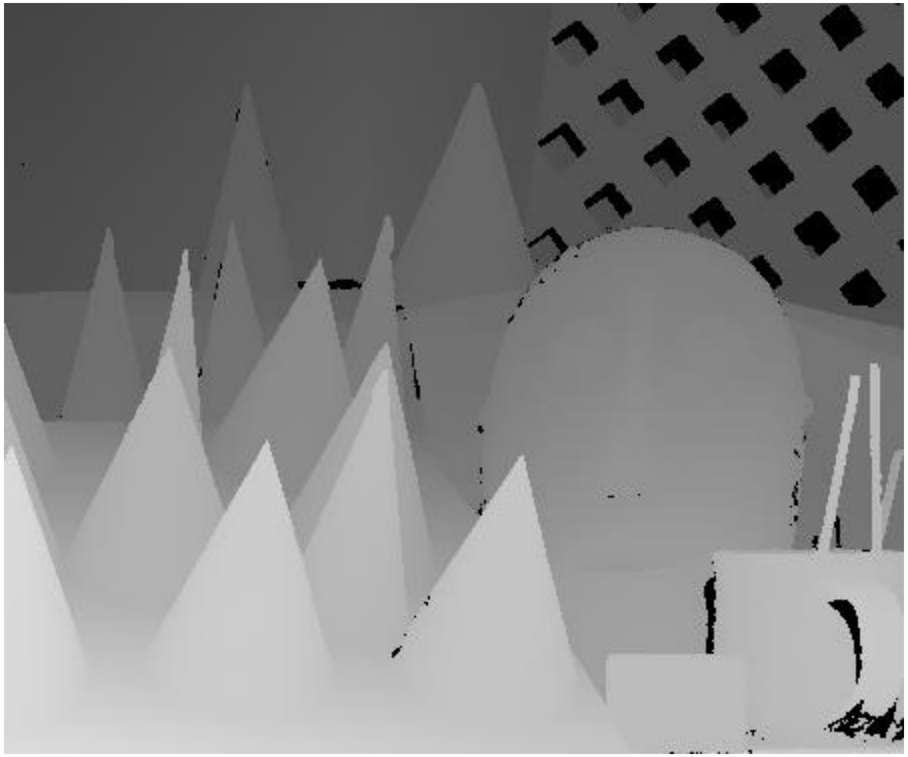
$$c_i = \beta\left(\frac{x - \hat{x}_{0i}}{\epsilon_x}\right)\beta\left(\frac{y - \hat{y}_{0i}}{\epsilon_y}\right),$$

$$\beta(\kappa) = \begin{cases} 2/3 - (1 - |\kappa|/2)\kappa^2, & \text{if } 0 \leq |\kappa| \leq 1 \\ (2 - |\kappa|)^3/6, & \text{if } 1 < |\kappa| < 2 \\ 0, & \text{otherwise,} \end{cases}$$

$\rightarrow$ $\mathbf{A}(\mathbf{x})\overrightarrow{\mathbf{p}}$

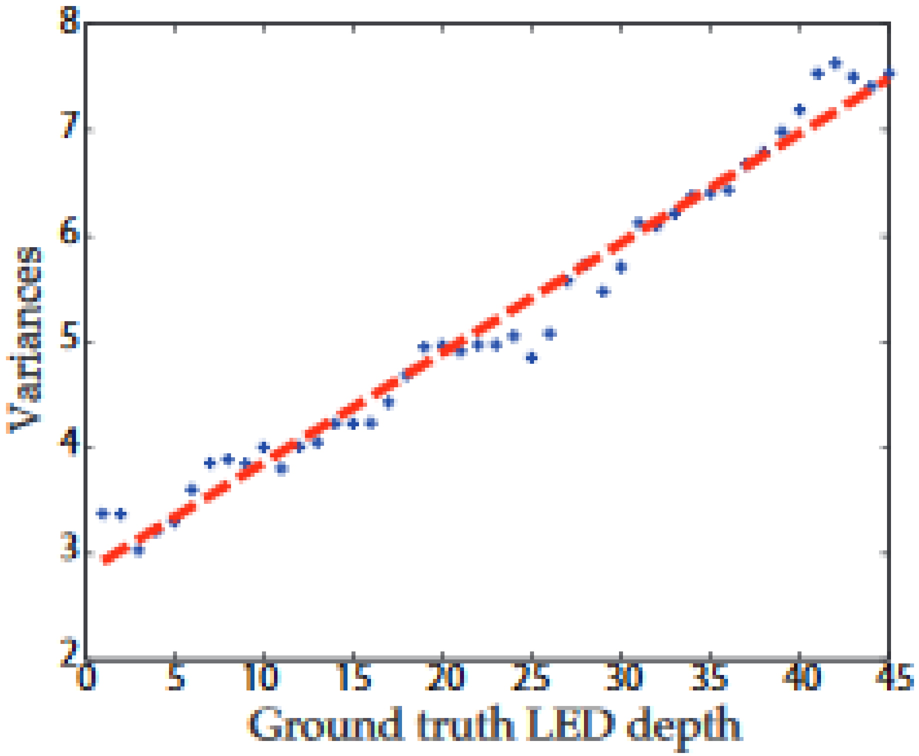
Take distance into consideration

- Using a distance map
- Calculate variance by distance of pixel



Take distance into consideration

- Based on the article's we read
- Liner connection between depth and var



Blur caused by turbulence

- The blur can be modeled by a overall optical transfer function (OTF)
- The OTF include the atmospheric OTF and the diffraction OTF.
- The overall OTF is giving by:

$$H(\rho) = H_{\text{atm}}(\rho) H_{\text{dif}}(\rho)$$

$$H_{\text{dif}}(\rho) = \begin{cases} \frac{2}{\pi}\left[\cos^{-1}\left(\frac{\rho}{2\rho_c}\right) - \frac{\rho}{2\rho_c}\sqrt{1-\left(\frac{\rho}{2\rho_c}\right)^2}\right], & \rho \le \rho_c \\ 0, & \text{otherwise} \end{cases},$$

$$H_{\text{atm}}(\rho) = \exp\left\{-3.44\left(\frac{\lambda l \rho}{r_0}\right)^{5/3}\left[1 - \alpha\left(\frac{\lambda l \rho}{D}\right)^{1/3}\right]\right\},$$

Blur caused by turbulence

- The overall OTF is giving by:

$$H(\rho) = H_{\text{atm}}(\rho) H_{\text{dif}}(\rho)$$

- The atmospheric OTF is giving by:

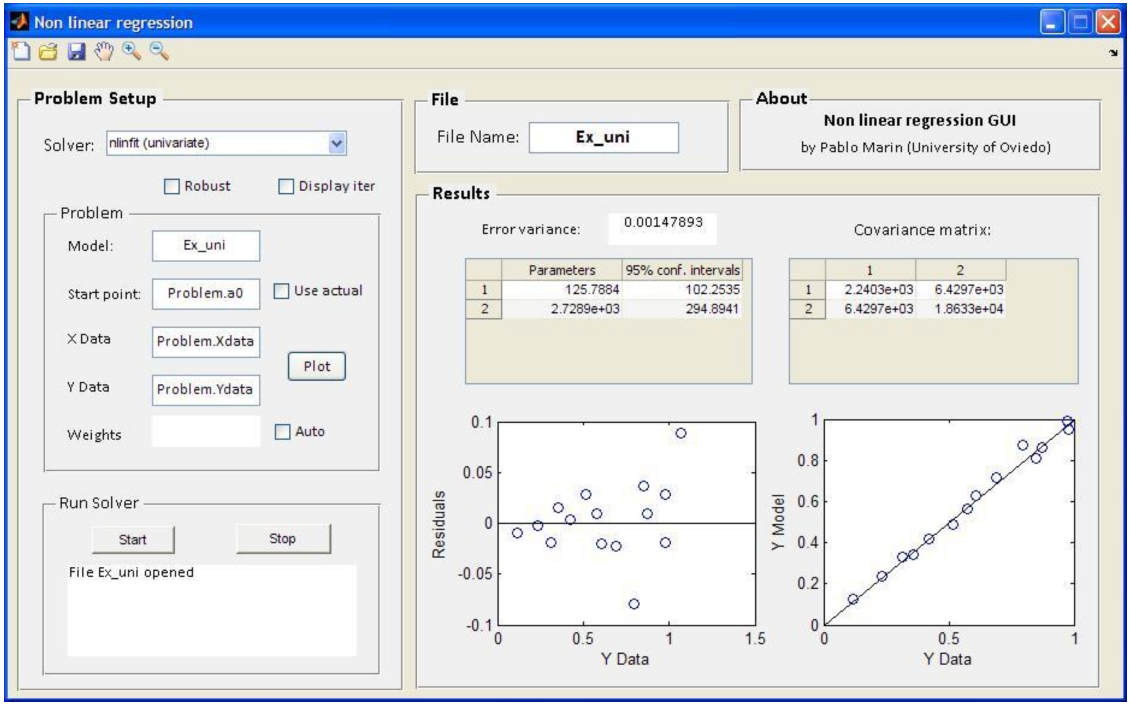
$$H_{\text{atm}}(\rho) = \exp\left\{-3.44\left(\frac{\lambda l \rho}{r_0}\right)^{5/3}\left[1 - \alpha\left(\frac{\lambda l \rho}{D}\right)^{1/3}\right]\right\},$$

- The diffraction OTF is giving by:

$$H_{\text{dif}}(\rho) = \begin{cases} \frac{2}{\pi}\left[\cos^{-1}\left(\frac{\rho}{2\rho_c}\right) - \frac{\rho}{2\rho_c}\sqrt{1-\left(\frac{\rho}{2\rho_c}\right)^2}\right], & \rho \leq \rho_c \\ 0, & \text{otherwise} \end{cases},$$

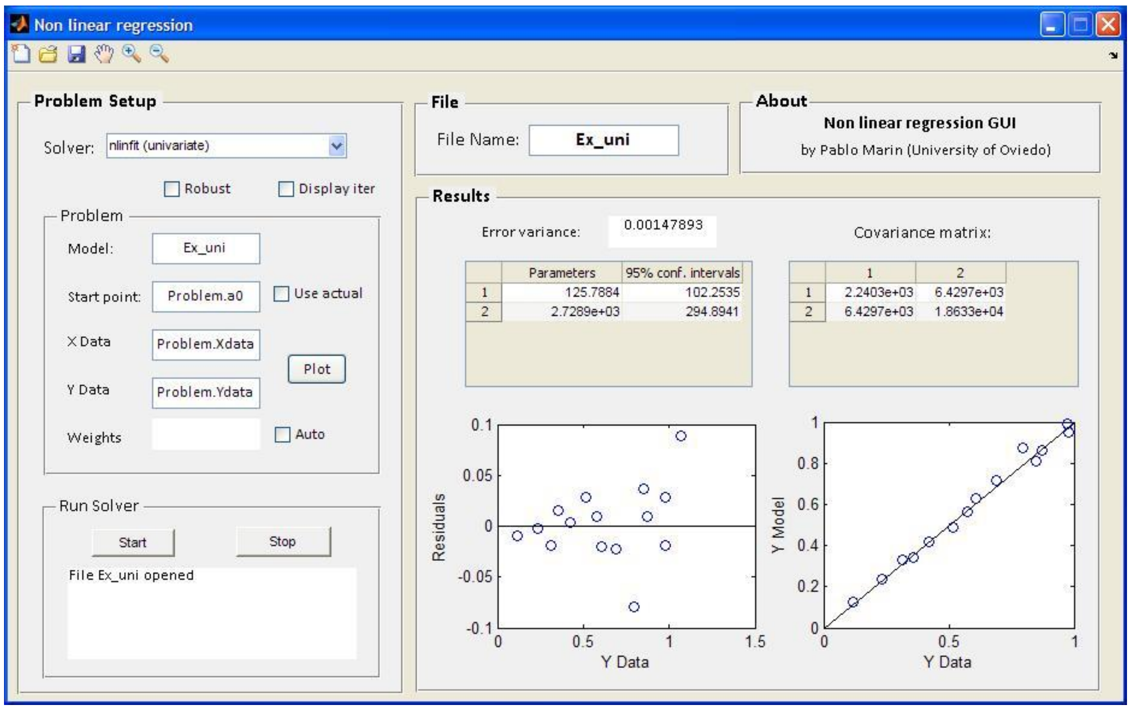
GUI

- We created simulator using GUI.
- Simulator tool for the user.
- Fast and convenient. (GUI VIDEO)



Problem with Time complexity

- The simulation took about 6 minutes.
- After some improvements – 4 seconds.
- 0.01% of the initial time.



Creating Video

- Considering all of the previous we created a video.
- Compering this video to real turbulence video provided by Rafael.

Time smoothing

- In order to create more "Realistic" videos we need to activate time smoothing on the video
- We used mean filter on the frames of the video

Results
distance effect – original Image

Results – distance effect (80m)

After - Cn^2=5e-16 ,f=1.2[m] ,D=0.2[m] Dist=80[m]

Results – distance effect (330m)

After - Cn^2=5e-16 ,f=1.2[m] ,D=0.2[m] Dist=330[m]

Results – distance effect (580m)

After - Cn^2=5e-16 ,f=1.2[m] ,D=0.2[m] Dist=580[m]

Results – distance effect (830m)

Results – distance effect (980m)

After - Cn^2=5e-16 ,f=1.2[m] ,D=0.2[m] Dist=980[m]

Results – distance effect (1590)

After - Cn^2=5e-16 ,f=1.2[m] ,D=0.2[m] Dist=1590[m]

Results – Cn effect
original image

Results – Cn effect –(1e-16)

After - Cn^2=1e-16 ,f=1.2[m] ,D=0.2[m]

Results – Cn effect –(2e-16)

After - Cn^2=2e-16 ,f=1.2[m] ,D=0.2[m]

Results – Cn effect –(3e-16)

After - Cn^2=3e-16 ,f=1.2[m] ,D=0.2[m]

Results – Cn effect –(4e-16)

After - Cn^2=4e-16 ,f=1.2[m] ,D=0.2[m]

Results – Cn effect –(5e-16)

After - Cn^2=5e-16 ,f=1.2[m] ,D=0.2[m]

Checking goodness

- Video from Raphael.
- Avg on video to get clear frame.
- Distort using our simulation.

Rafael Video – dis 1500m

Simulated video

Quantitative Comparison

- Using sophisticated algorithm for tracking pixels.
- Calculating var of pixel movements.
- Var for real video : var.x= 0.301 var.y =0.441
- Var for simulate video :
  var.x= 0.175 var.y=0.233

- Results are fantastic!

Summary

- We have managed to apply “statistic” turbulence.
- We will apply a physical model in the future.
- We will create a GUI.
- If time allows – we’ll make the system capable of handling with videos and apply blur.

Conclusions

- We have learned that as the number of reference points grows up (Milanfar), the image distortion level grows.
- Milanfar solution is a good statistical solution.
- In the future – simulate more physical distortion for better results.

References

- [1] Endre Repasi and Robert Weiss , "Computer Simulation of Image Degradations by Atmospheric Turbulence for Horizontal Views" , 2011
- [2] Peyman Milanfar , "Removing Atmospheric Turbulence via Space-Invariant Deconvolution" , 2013