GIST: Context Based Scene Recognition

Rahul Sawhney
Abhijit Kundu

GIST

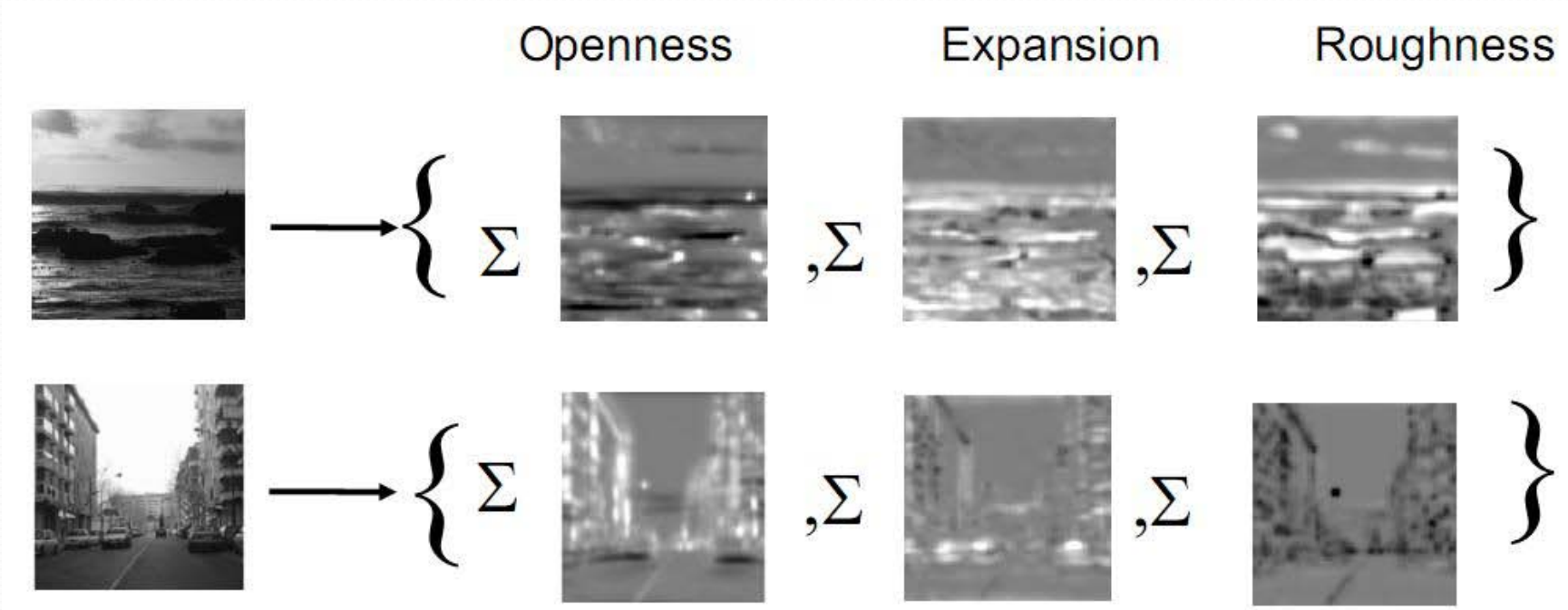
- Gist is a holistic statistical signature of the image, yielding abstract scene classification and layout.

Forests are “enclosed”

Beaches are “open”

GIST: Spatial Envelope

- A scene can be represented by a vector of values for each spatial envelope properties.



GIST: Application & Advantages

- Considers the whole scene. Not based on detection of individual objects in the scene.
- Can be used as a prior for object detection, i.e it can improve object detection based on the context of the scene.

(a) Isolated object (b) Object in context (c) Low-res Object

GIST: the method

- Create filter bank: precompute the filter transfer functions(gabor filters), tuned to different orientations and scale.

- Prefilter the image: normalization of luminance variance, contrast, whitening transform. Costly operations include FFT, padding.

- Compute Gist Descriptor: Convolve the filters computed earlier in frequency domain, so that convolution is element-wise matrix multiplication. Divide the image into local regions and the mean of magnitude of local features represent the gist descriptor.

- Dimension Reduction: Default configuration of 3 scales, with 8, 8, and 4 orientations, for 4x4 windows computed in 3 channel color image gives 960 features. Methods like PCA, ICA are used to reduce dimensions.

GIST models

- [1]. Oliva & Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope, ” IJCV 2001 (MATLAB code for computing gist feature vectors available (link))
- [2]. Torralba et. al. , “Context-based vision system for place and object recognition,” ICCV 2003
- [3]. Renninger & Malik, “When is scene identication just texture recognition?”, Vision Research 2004
- [4]. Siagian & Itti, “Rapid biologically-inspired scene classification using features shared with visual attention,” PAMI 2007 (C++ implementation integrated into iLab Neuromorphic Vision C++ Toolkit(link))

GIST: Oliva & Torralba (IJCV 2001)

- The model computes magnitude spectrum of Windowed Fourier Transform, here we use 4 by 4 equally-sized, non-overlapping regions. We then reduce the feature dimension with Principal component Analysis (PCA) before classifying the scenes.

- Matlab code to compute gist descriptor available.(link)

- We based our implementation over this model.

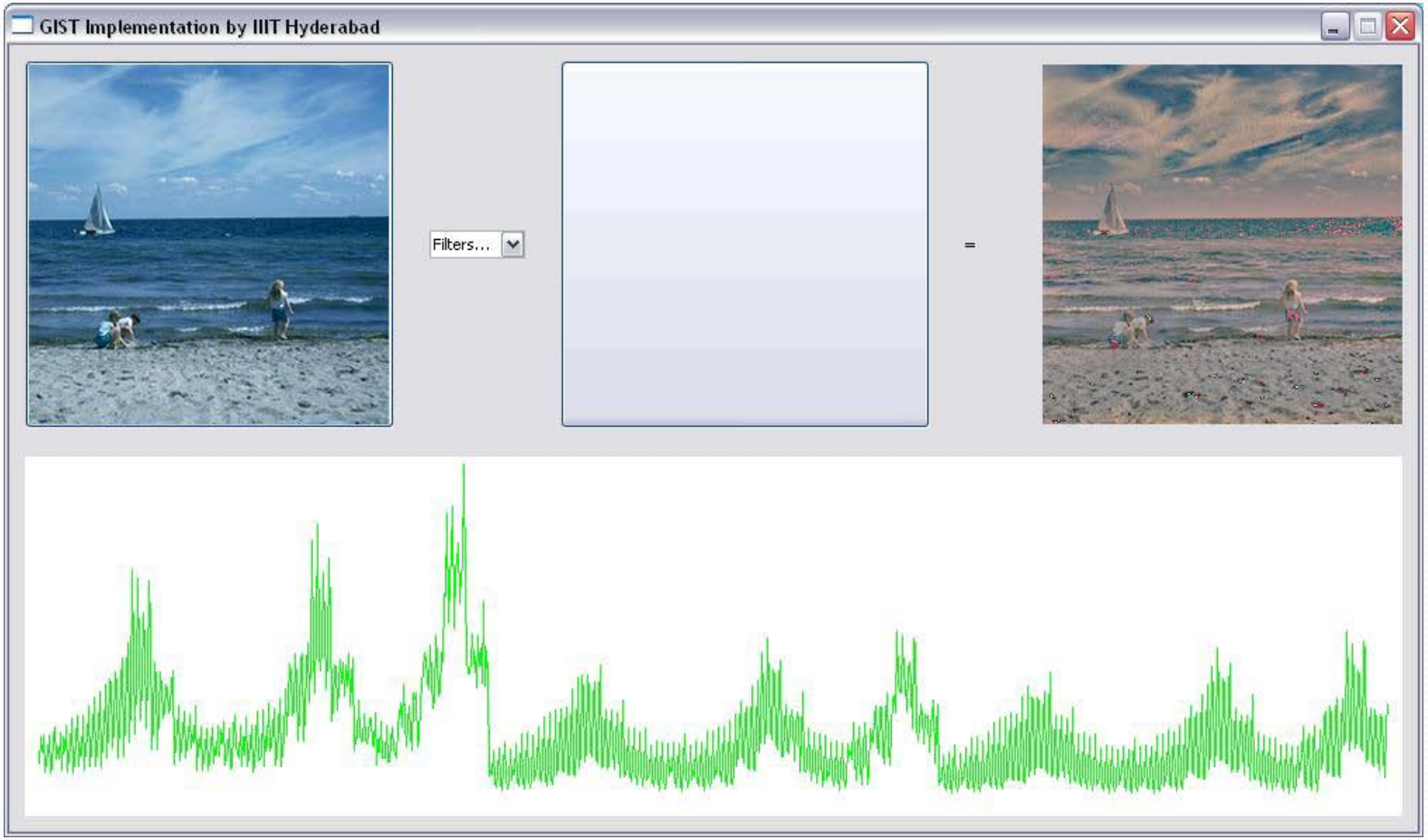
GIST: Our implementation

1. **Matlab code:** Available online for computing gist descriptor of a scene. (Olivia-Torallba, 2001)

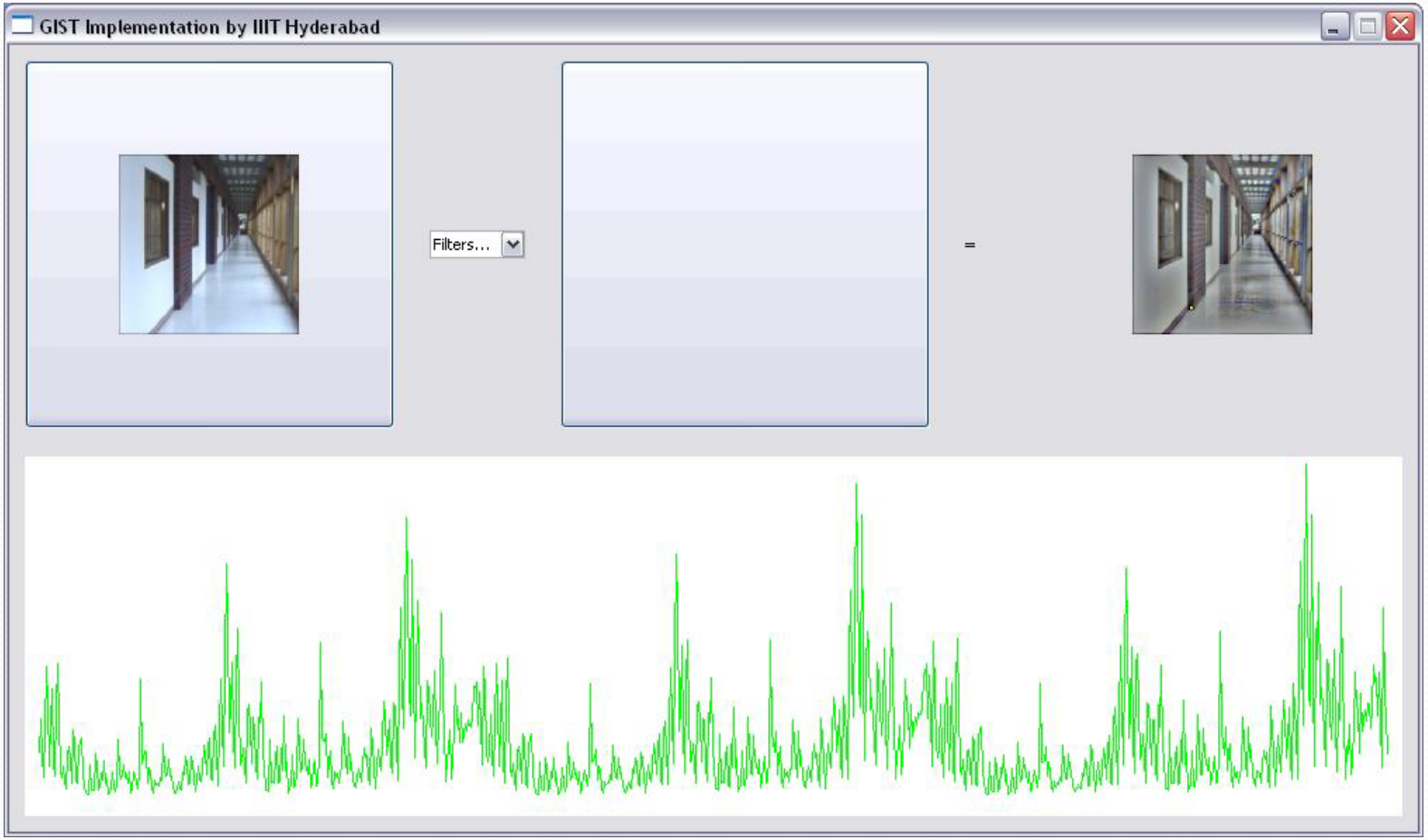
2. **C implementation:** We implemented the torallba 2001 gist model (in matlab) in C. We used FFTW libraries to compute fourier transform. Slightly faster than matlab.

3. **GPU parallelization:** Some of the functions has been parallelized by running them in GPU using CUDA. We also make use of the inbuilt libraries like CUFFT & CUBLAS.

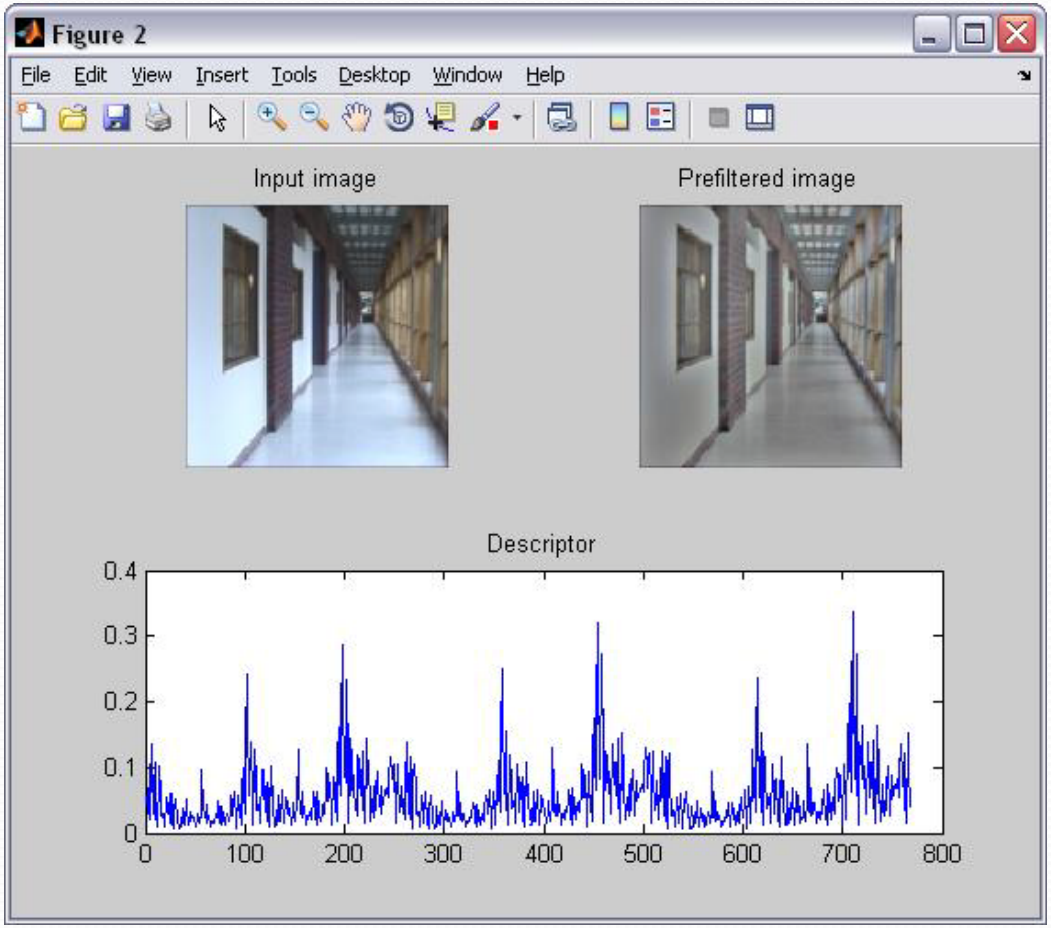
GIST: Results



GIST: Results



GIST: Results

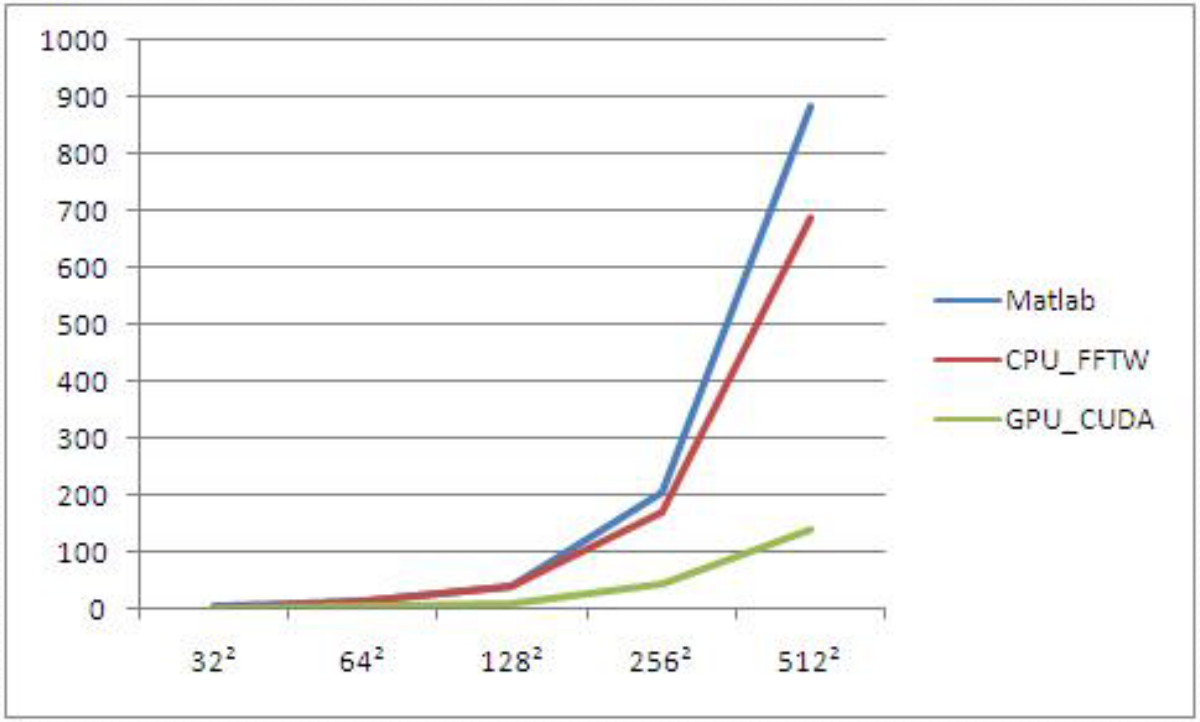


GIST: GPU Parallelization

- All the operations performing on each pixels can be parallelized.
- Each thread can compute the operations on each of these pixels. Thus total number of threads can go up to (img_size x img_size x channels).
- Use inbuilt libraries to calculate transforms like FFT, available in CUFFT.
- Convolution is done in frequency domain, and is faster than naïve convolution. Again use of FFT.

GIST: Comparison

Time chart for computing the function to create Gabor filter bank



32^2 indicates an image of 32x32 pixels, Time is shown in milliseconds

Future Work

- Parallelize more portion of the code. Use of advanced features like thread cooperation, shared memory…

- Make OpenCV implementation of GIST (need to use the API of opencv, DFT functions exists in opencv. Also need to follow OpenCV coding style and guidelines.

GIST: Context Based Scene Recognition

?