Compressed Dynamic Mode Decomposition for Background Modeling

Erichson, Brunton, and Kutz

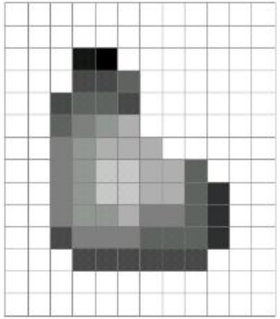
Wenhao Wang, Leo Orozco, Josh Myers-Dean

Overview

- Image/Video Processing Background
- Motivation and Challenges for Background Modeling
- Review of DMD
- Methodology
- Results
- Tutorial

Image Processing Background: What is an Image?

- Focus on **grayscale** images
- Represent an image as 2D grid of numbers
- Each pixel in [0, 255] - intensity
    - Usually normalize to [0, 1] for precision



=

255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	20	0	255	255	255	255	255	255	255
255	255	255	75	75	75	255	255	255	255	255	255
255	255	75	95	95	75	255	255	255	255	255	255
255	255	96	127	145	175	255	255	255	255	255	255
255	255	127	145	175	175	175	255	255	255	255	255
255	255	127	145	200	200	175	175	95	255	255	255
255	255	127	145	200	200	175	175	95	47	255	255
255	255	127	145	145	175	127	127	95	47	255	255
255	255	74	127	127	127	95	95	95	47	255	255
255	255	255	74	74	74	74	74	74	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255

Image: https://facultyweb.cs.wwu.edu/~wehrwes/courses/csci497p_20f/lectures/L01.pdf

Image Processing Background: What is an Image?

- Row-column indexing
- Formally, intensity is a function of space $f : \mathbb{R}^2 \to \mathbb{R}$

Row

Col

$\rightarrow$

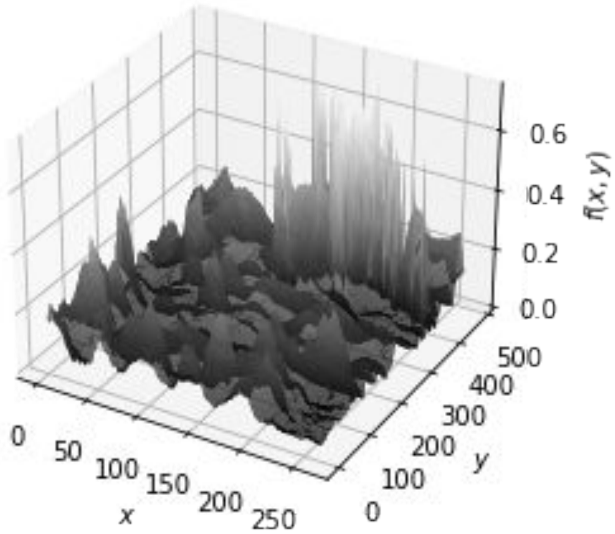


Image Processing Background: What is a Video?

- Tensor of Images
  - Channel dimension is time
- Can also represent as a matrix
  - Overdetermined

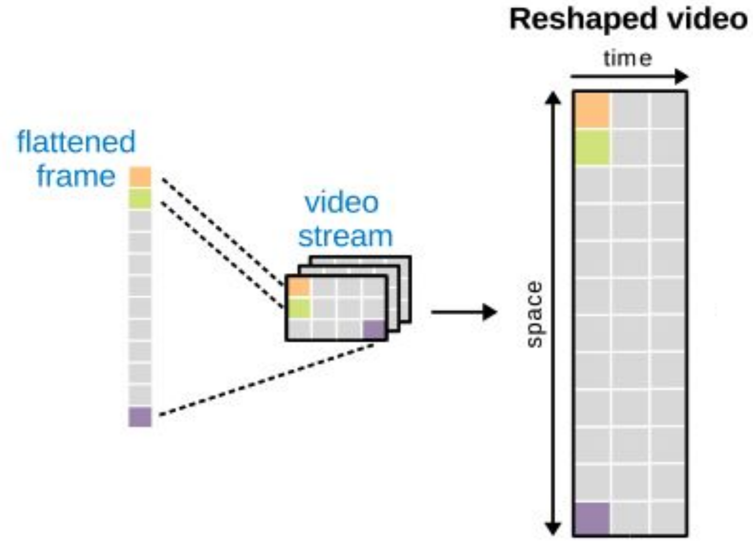


Image: Figure 1

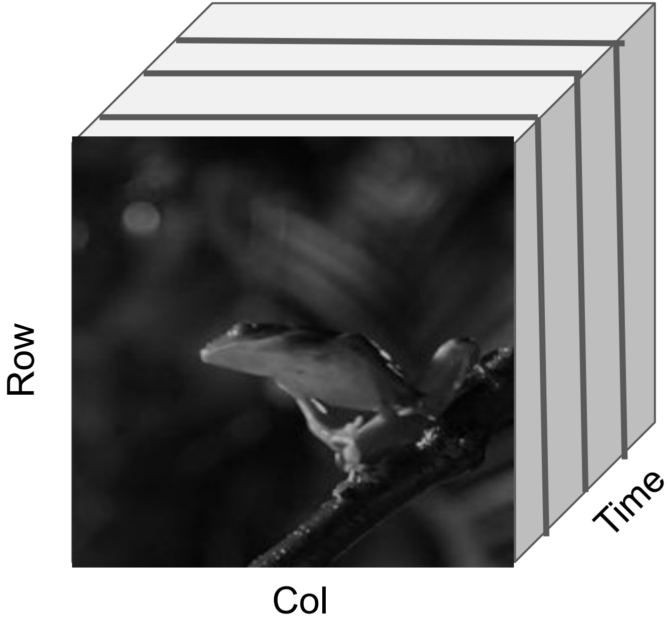
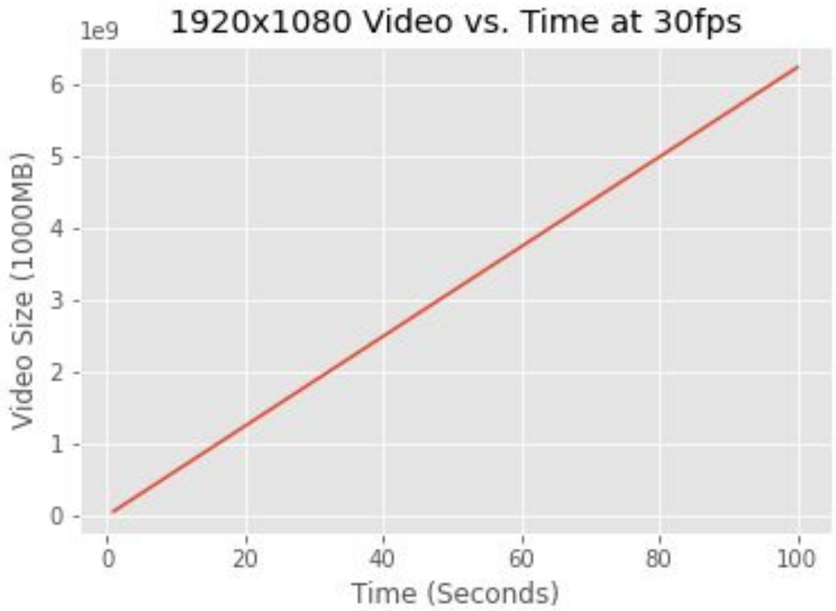


Image Processing Background: Considerations

- Algorithms need to be **efficient**
- Consider a 1920x1080 image
  - One byte per pixel ~ 2MB image
- Video: 10 seconds at 30fps? ~600MB



Overview

- Image/Video Processing Background
- Motivation and Challenges for Background Modeling
- Review of DMD
- Methodology
- Results
- Tutorial

Motivation of Background Modeling

- Modeling background allows us to isolate foreground (segmentation)
- Track objects (surveillance)
- Replace background (image editing)

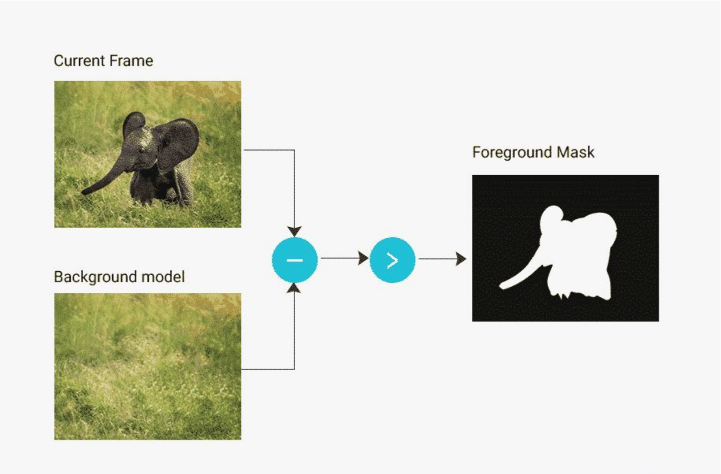


Image: OpenCV

Image: WEHRWEIN AND SZELISKI: VIDEO SEGMENTATION WITH BACKGROUND MODELS

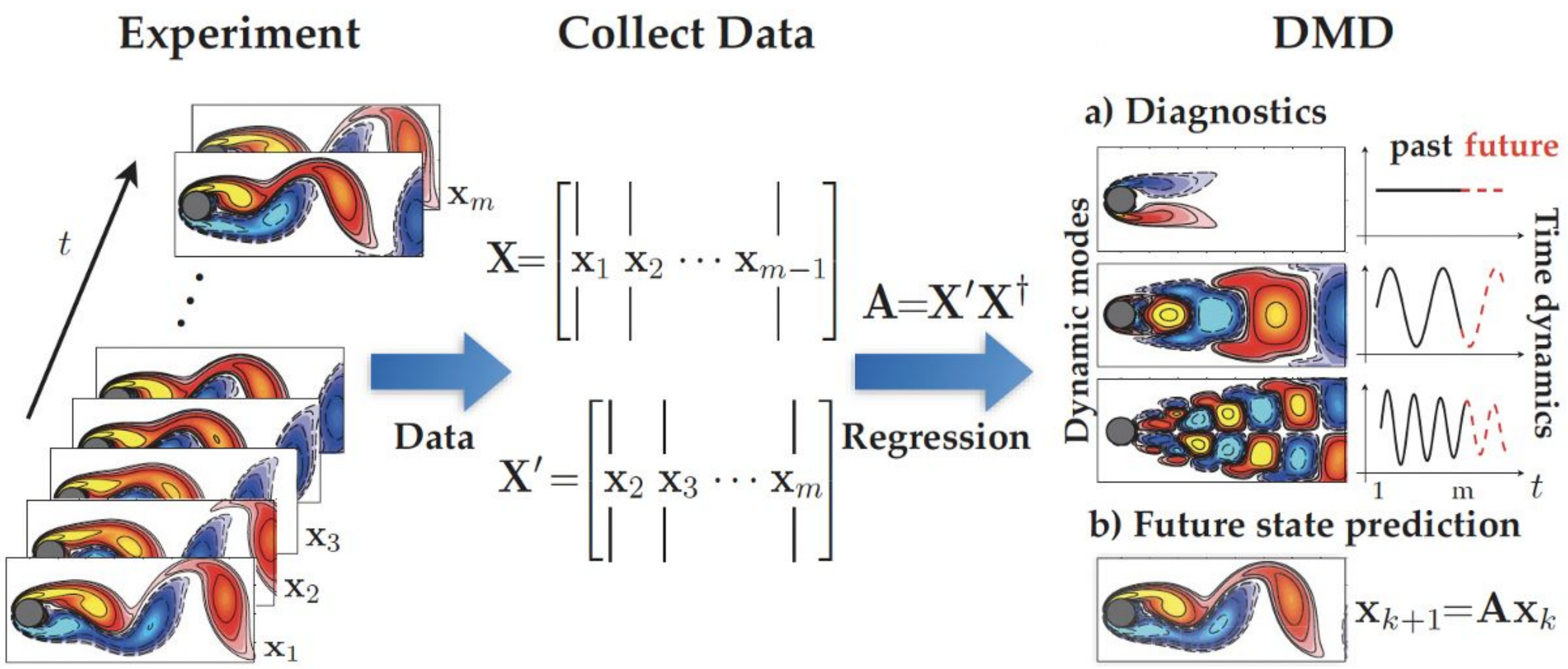
Challenges in Background Modeling

- Illumination (e.g., fog, sun)
- Noise (e.g., bad weather)
- Changes in camera pose (e.g., moving camera)
- Objects leaving/entering frame (e.g., sudden occlusions)
- Sleeping foreground objects (e.g., parking)

Overview

- Image/Video Processing Background
- Motivation and Challenges for Background Modeling
- Review of DMD
- Methodology
- Results
- Tutorial

Review of Dynamic Mode Decomposition(DMD)



Overview

- Image/Video Processing Background
- Motivation and Challenges for Background Modeling
- Review of DMD
- Methodology
- Results
- Tutorial

DMD in image processing

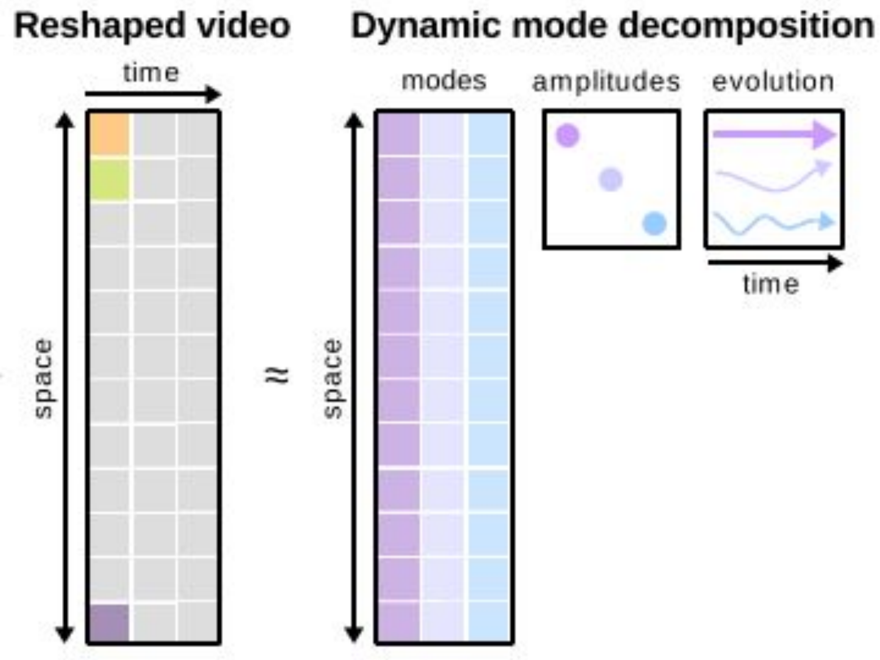
$$\mathbf{X}' \approx \mathbf{A}\mathbf{X}.$$

$$\mathbf{A} = \underset{\mathbf{A}}{\operatorname{argmin}} \|\mathbf{X}' - \mathbf{A}\mathbf{X}\|_F = \mathbf{X}'\mathbf{X}^{\dagger}$$

$$\boldsymbol{\Phi} = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\boldsymbol{\Sigma}}^{-1}\mathbf{W}.$$

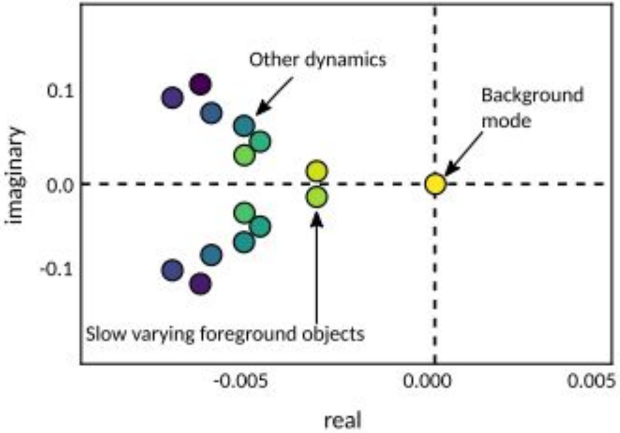
$$\boldsymbol{\Phi}\mathbf{B}\mathcal{V} = \begin{pmatrix} \phi_{11} & \phi_{1p} & \cdots & \phi_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{i1} & \phi_{ip} & \cdots & \phi_{ik} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{n1} & \phi_{np} & \cdots & \phi_{nk} \end{pmatrix} \begin{pmatrix} b_1 & & & & \\ & \ddots & & & \\ & & b_p & & \\ & & & \ddots & \\ & & & & b_k \end{pmatrix}$$

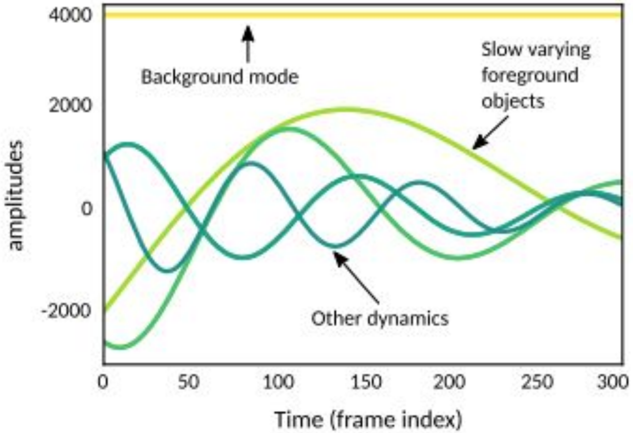
$$\times \begin{pmatrix} 1 & \lambda_1 & \cdots & \lambda_1^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_p & \cdots & \lambda_p^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_k & \cdots & \lambda_k^{m-1} \end{pmatrix},$$



DMD in image processing

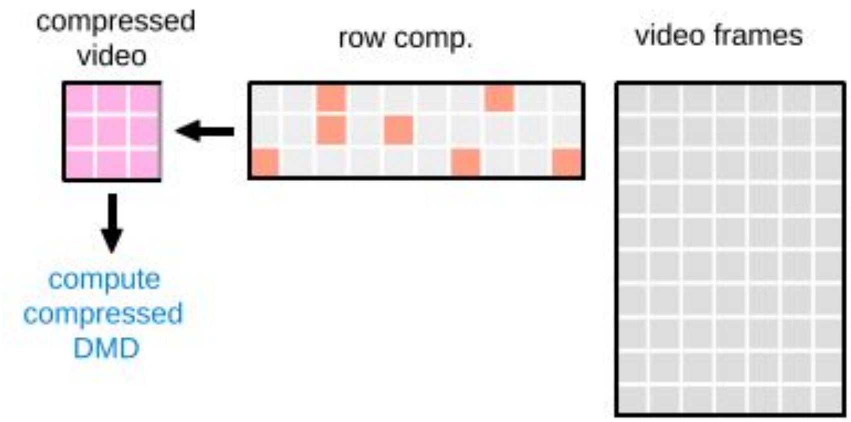
(a)

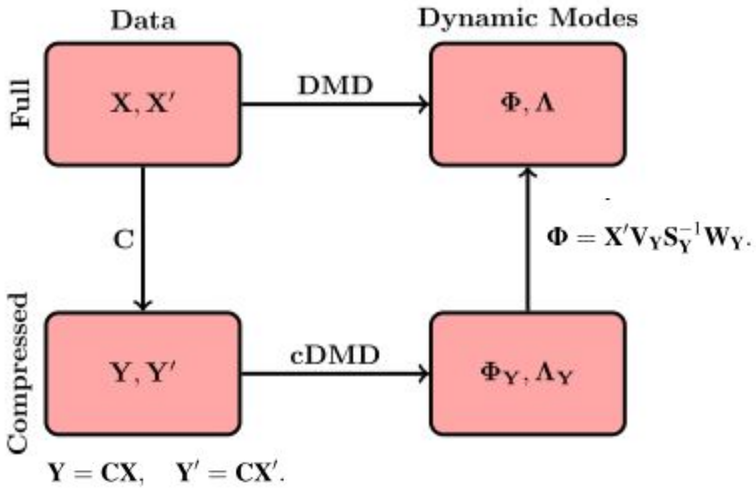




Compressed Dynamic Mode Decomposition (cDMD)

- DMD is computed on sparse measurements without using full data
- Computationally efficient





Methods

$$\tilde{\mathbf{x}}_t = \sum_{j=1}^{k} b_j \phi_j \lambda_j^{t-1}.$$

function $[\boldsymbol{\Phi}, \mathbf{b}, \mathcal{V}] = \texttt{cdmd}(\mathbf{D}, k, p)$

(1)	$\mathbf{X}, \mathbf{X}' = \mathbf{D}$	Left/right snapshot sequence.
(2)	$\mathbf{C} = \texttt{rand}(p, m)$	Draw $p \times m$ sensing matrix.
(3)	$\mathbf{Y}, \mathbf{Y}' = \mathbf{C} * \mathbf{D}$	Compress input matrix.
(4)	$\mathbf{U}, \mathbf{S}, \mathbf{V} = \texttt{svd}(\mathbf{Y}, k)$	Truncated SVD.
(6)	$\tilde{\mathbf{A}} = \mathbf{U}^* * \mathbf{Y}' * \mathbf{V} * \mathbf{S}^{-1}$	Least squares fit.
(7)	$\mathbf{W}, \boldsymbol{\Lambda} = \texttt{eig}(\tilde{\mathbf{A}})$	Eigenvalue decomposition.
(8)	$\boldsymbol{\Phi} \leftarrow \mathbf{X}' * \mathbf{V} * \mathbf{S}^{-1} * \mathbf{W}$	Compute full-state modes $\boldsymbol{\Phi}$.
(9)	$\mathbf{b} = \texttt{lstsq}(\boldsymbol{\Phi}, \mathbf{x}_1)$	Compute amplitudes using $\mathbf{x}_1$ as intial condition.
(10)	$\mathcal{V} = \texttt{vander}(\texttt{diag}(\boldsymbol{\Lambda}))$	Vandermonde matrix (optional).

$$\begin{aligned}\mathbf{X}_{\mathbf{DMD}} &= \mathbf{L} + \mathbf{S} \\ &= \underbrace{\sum_{p} b_p \phi_p \lambda_p^{t-1}}_{\text{Background Video}} + \underbrace{\sum_{j \neq p} b_j \phi_j \lambda_j^{t-1}}_{\text{Foreground Video}}\end{aligned}$$

Overview

- Image/Video Processing Background
- Motivation and Challenges for Background Modeling
- Review of DMD
- Methodology
- Results
- Tutorial

GPU accelerated implementation

- CPU: Very low number of ALU which are optimized for low latency cached data sets
- GPU: several small ALU optimized for data parallel high throughput computations

*ALU = Arithmetic Logic Units

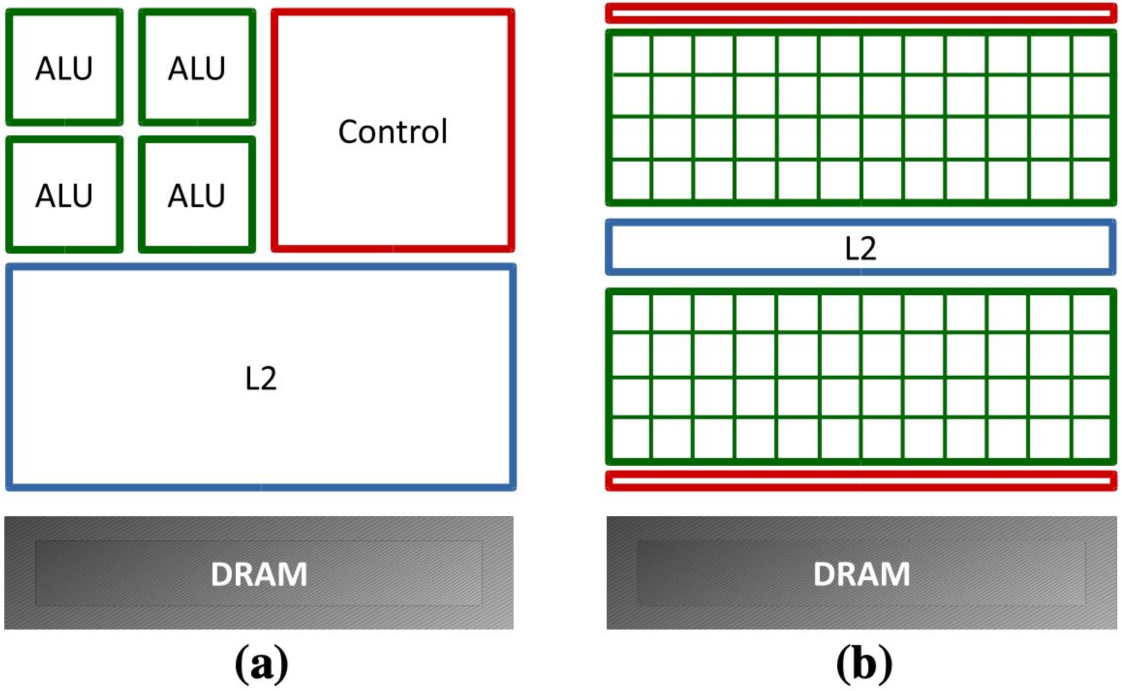


Fig. 5 Illustration of the CPU and GPU architecture. **a** CPU. **b** GPU

Key takeaways from implementation of accelerated GPU vs. accelerated CPU:

- GPU-accelerated DMD is substantially faster than the traditional MKL (Intel Math Kernel Library) accelerated routine.

- Sparse cDMD algorithm performed slightly faster than the Exact DMD algorithm; due to the implicit regularization of random algorithms.

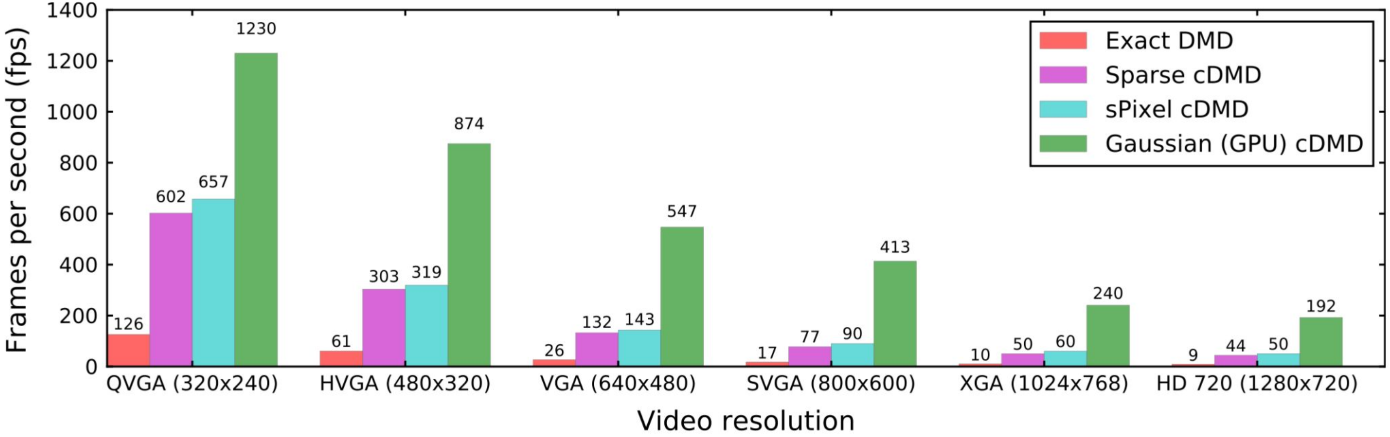


Figure 12.

Panel of performance of the sparsity-promoting compressed DMD algorithm

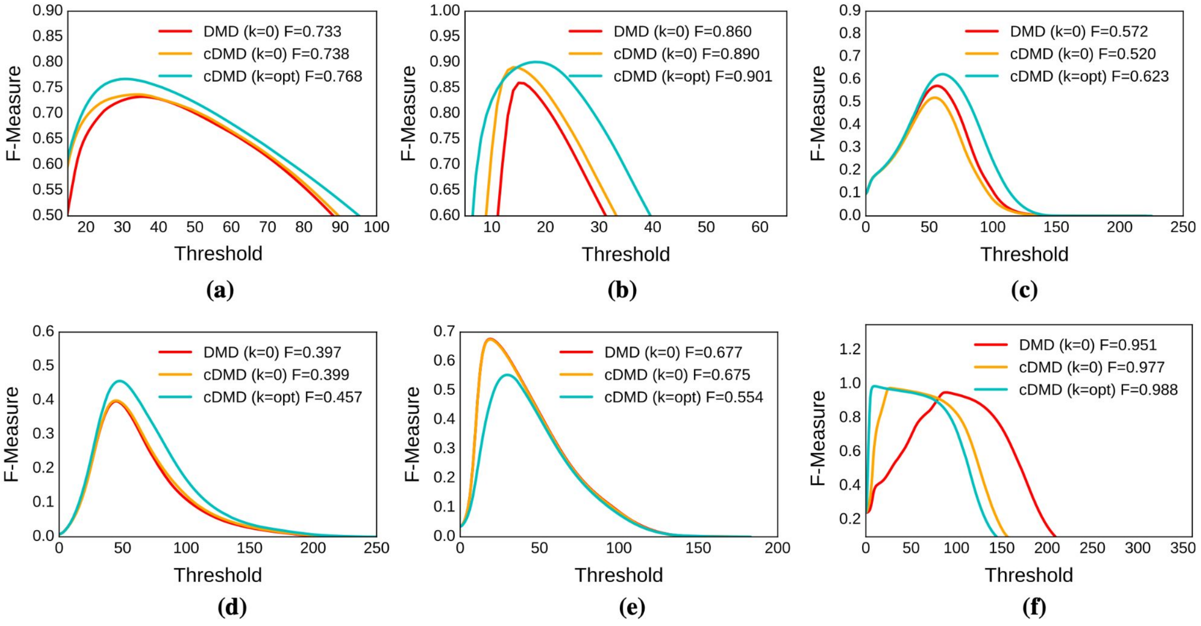


Figure 8

- F- measure represents the harmonic mean of:
  - Precision: # of predicted pixels which are actually correct.
  - Recall: The ability of the algorithm to predict the foreground pixels belonging to moving objects.
- Threshold:
  - Cutoff range of background and foreground boundaries.

Qualitative results

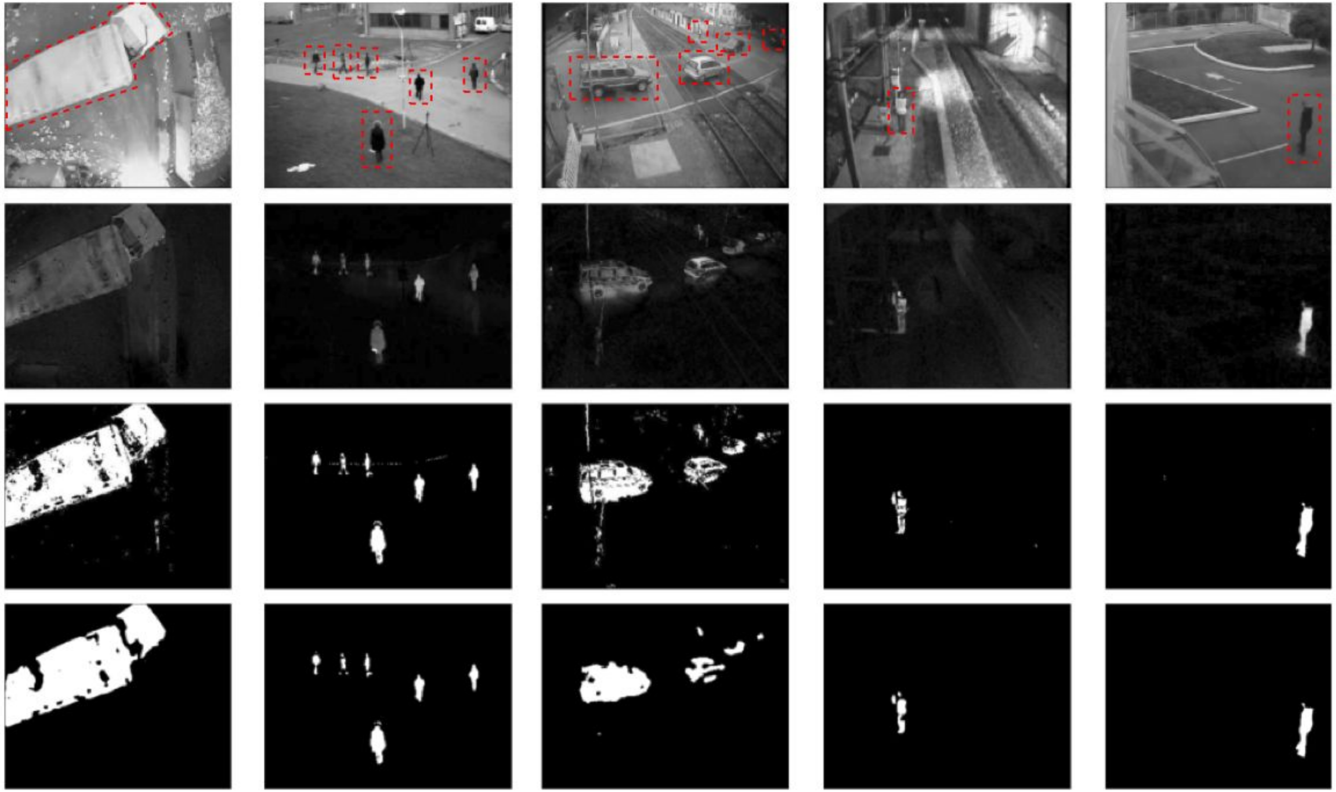


Figure 9

Main takeaways from the paper:

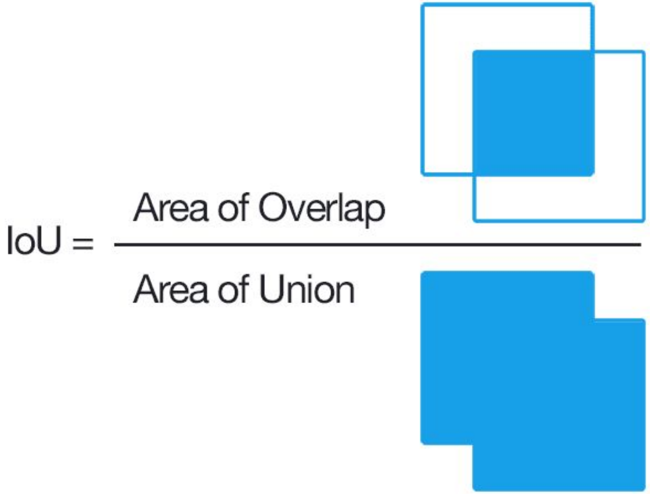
- GPU-accelerated DMD is substantially faster than the traditional MKL (Intel Math Kernel Library) accelerated routine.
- Varying # of modes in background modeling can yield more accurate background models - rather than just using the zero mode
- cDMD is much faster than Exact DMD without sacrificing performance

Overview

- Image/Video Processing Background
- Motivation and Challenges for Background Modeling
- Review of DMD?
- Methodology
- Results
- Tutorial

Tutorial

- Go to https://github.com/joshmyersdean/DMD_Background_Modeling
  - Will paste in chat
- Click on dmd.ipynb -> open in colab
  - Instructions also in README
- Places to change listed in README



Source: pyimagesearch