Machine Learning Lunch

Parsimonious Linear Fingerprinting for Time Series

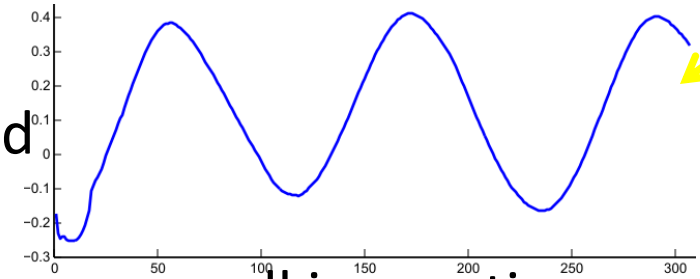
Lei Li
joint work with B. Aditya Prakash, Christos Faloutsos
School of Computer Science
Carnegie Mellon University

Why study time series?

Motion Capture

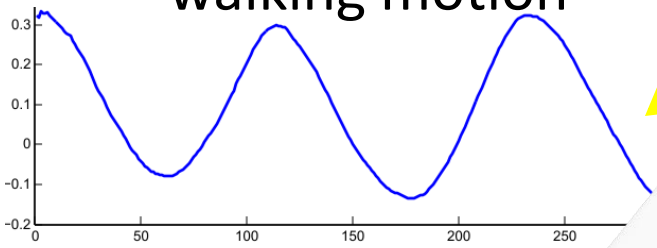
- Synthesize human motion
- Design robots to assist the disabled

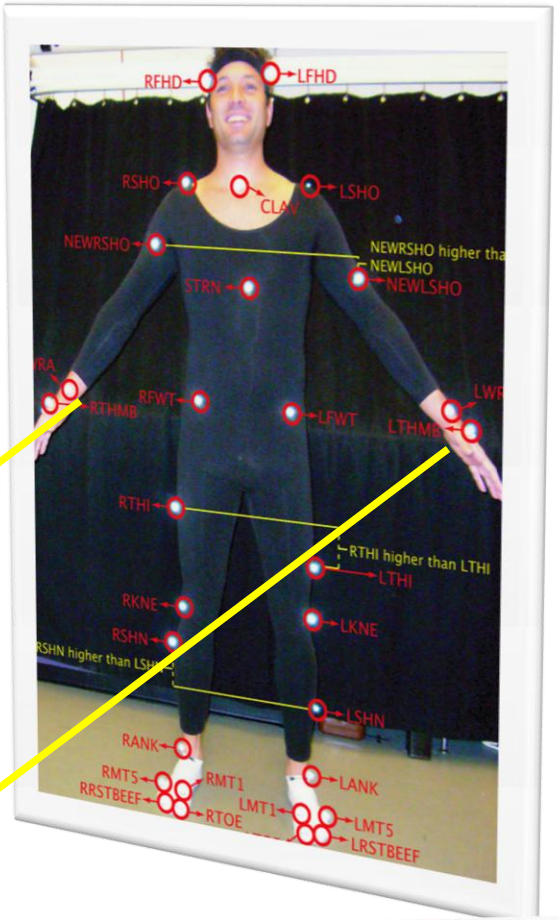
Right hand



walking motion

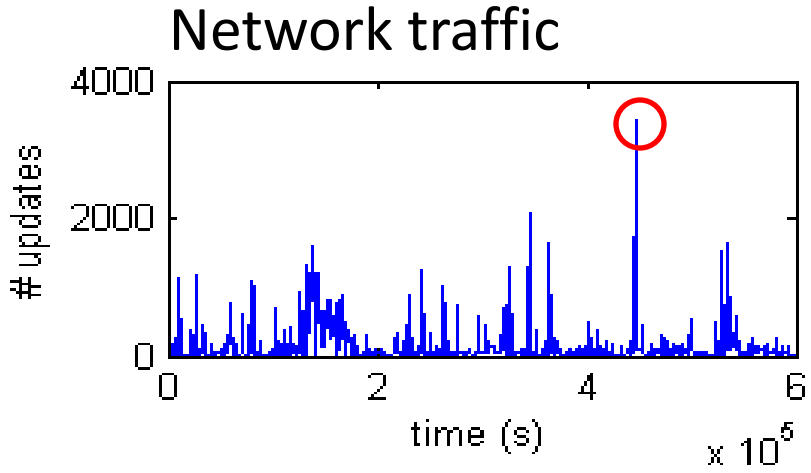
Left hand





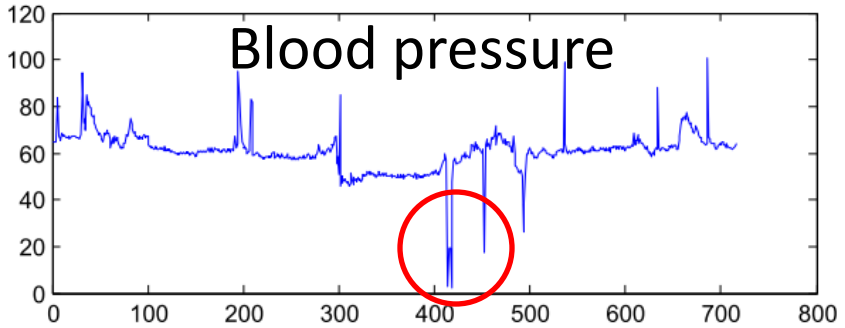
Network Security

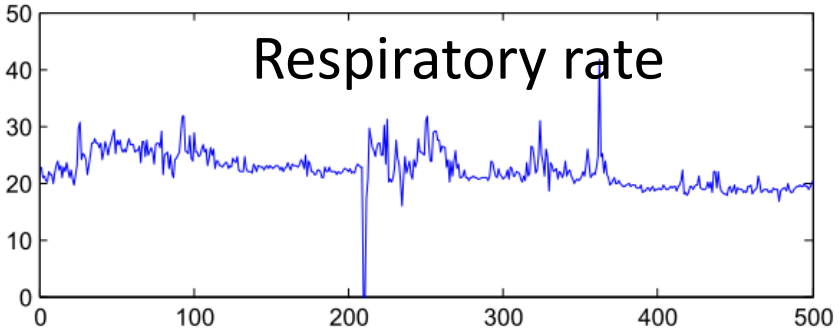
- Anomaly detection in computer network



Healthcare

- Monitoring physiologic signals at ICU
- Help early detection of fatal event using forecasting, classification/clustering

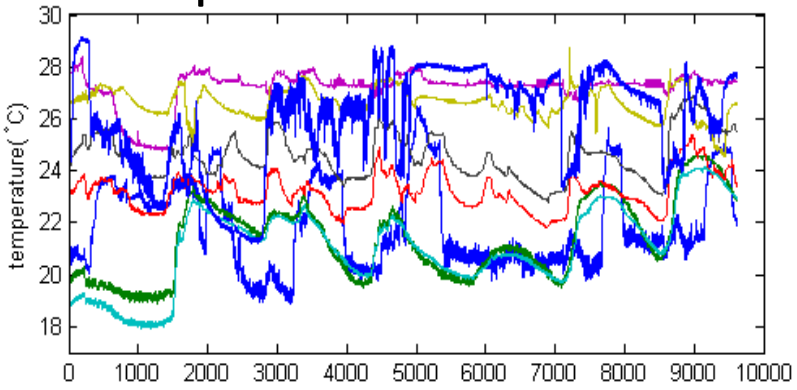




Datacenter Monitoring

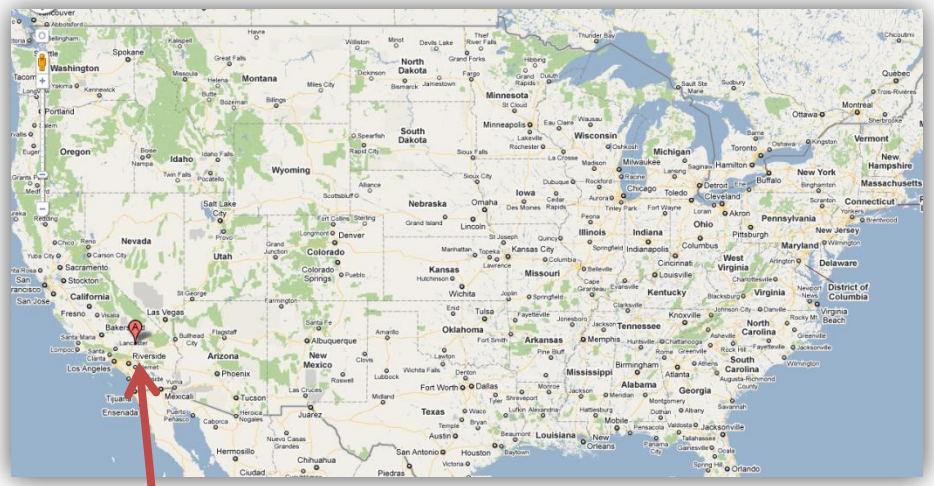
- Monitoring a datacenter with 5000 servers: 1TB data per day, 55 million streams ([Reeves+ 2009])
- Goal: save energy in data centers
  - US alone, $4.5billion power consumption (2006)

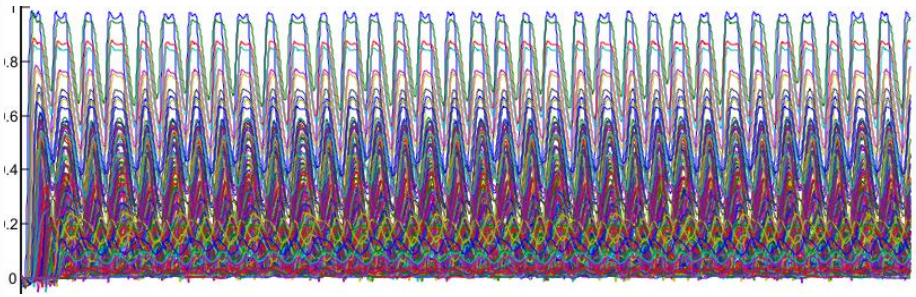
Temperature in datacenter



Environmental Monitoring

- Chlorine sensor in drinking water systems





Barstow residents advised not to drink tap water because of possible contamination

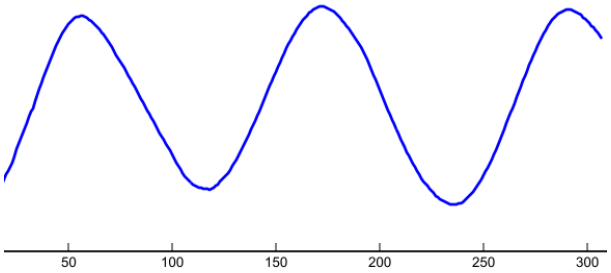
November 19, 2010 | 5:54 pm

Barstow residents were warned Friday not to drink local tap water, which recent tests show may be contaminated.

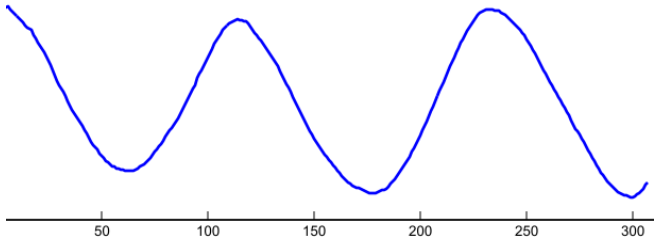
Golden State Water Co. posted the warning on its website after the agency was notified Thursday that samples contained high levels of perchlorate, an inorganic chemical that interferes with the human thyroid gland, affecting hormones as well as prenatal growth during pregnancies.

Central Problem

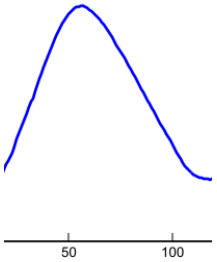
- Estimate "Similarity" among time sequences



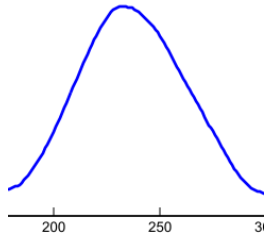
Are they Similar ?



Extract features



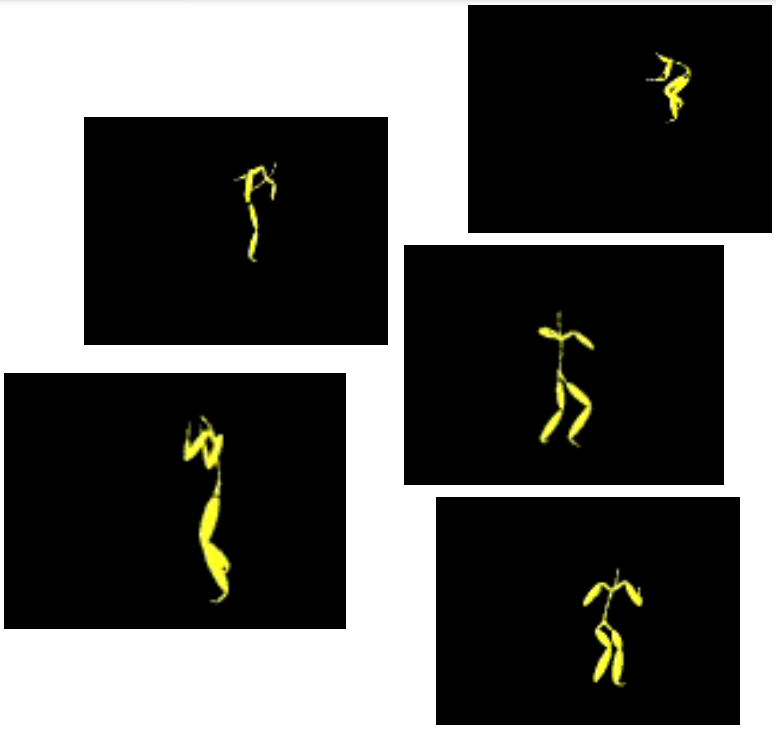
Distance(• , •)



Features (e.g. average, Fourier)

features

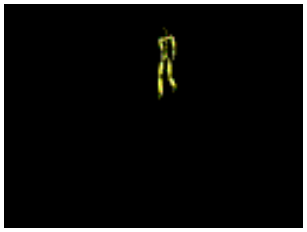
With similarity function: find the most similar motion sequence



SELECT * **FROM**

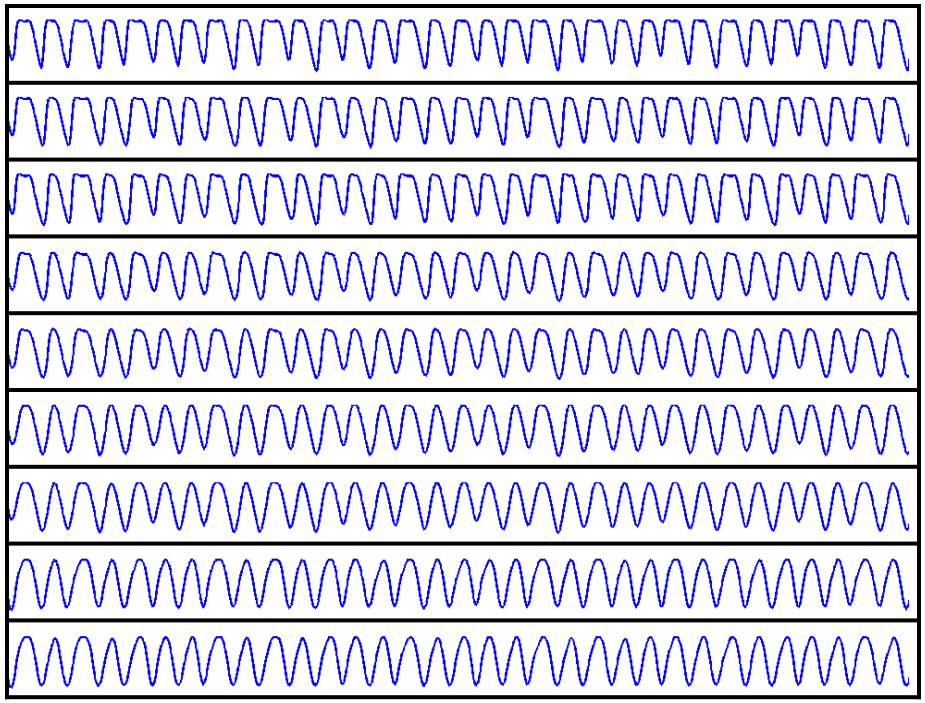
WHERE time_seq.

LIKE

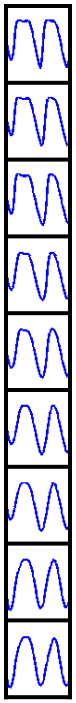


Underlying Question

What are good features / “fingerprints”?



e.g. length=4k,
in Chlorine measurement



Requirements of good features:

1. Time Shift
2. Frequency Proximity
3. Grouping Harmonics

Benefits

Good features / “fingerprints” help with

- Answering similarity queries
- Clustering/Classification
- Compression
- Forecasting

Outline

- Motivation & Problem Definition
- An Obvious Solution & Related Work
- Proposed Method: Intuition & Example
- Experiments & Results
- PLiF: High Level Ideas
- PLiF: Low Level Details
- Conclusion

Related Work

- Time series indexing [Keogh et al 02,04, …]
- Distance function
  - Euclidean distance [Rafiei et al 97, Ogras et al 06, …]
  - Dynamic time warping [Fu et al 05, Keogh 02, …]
- Fourier / Wavelets [Gilbert et al 01, Jahangiri et al 01, …]
- Dimensionality Reduction
  - PCA / SVD [Jolliffe 86]
  - ICA [Hyvarinen et al 01]

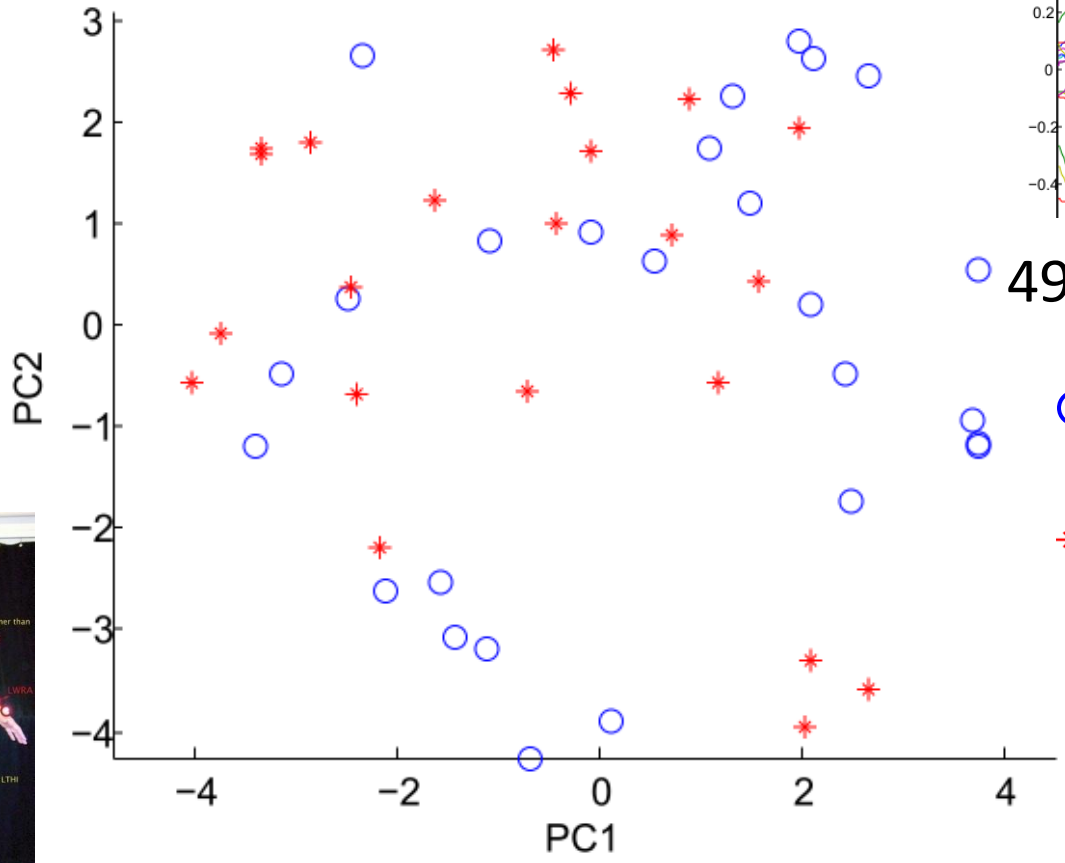
An Obvious (but failed) Solution

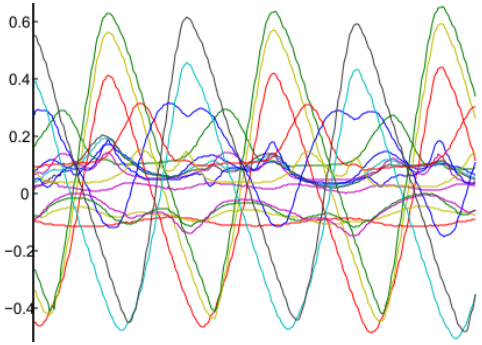
- Principal Component Analysis (PCA) / SVD
  - Dimensionality reduction
  - Very effective in many cases with high dimensional data
  - “Swiss army knife” in linear algebra

Solved?

Let us try the “Swiss Army Knife”

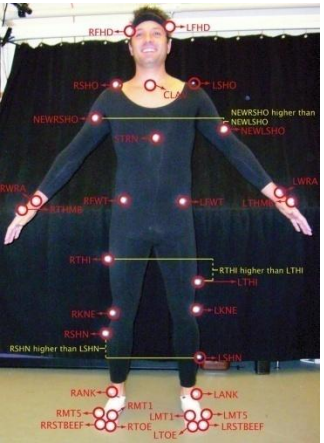
PCA gets confused





49 motion sequences

- walking motion
- running motion



Outline

- Motivation
- An Obvious Solution & Related Work
- Proposed Method: Intuition & Example
- Experiments & Results
- PLiF: High Level Ideas
- PLiF: Low Level Details
- Conclusion

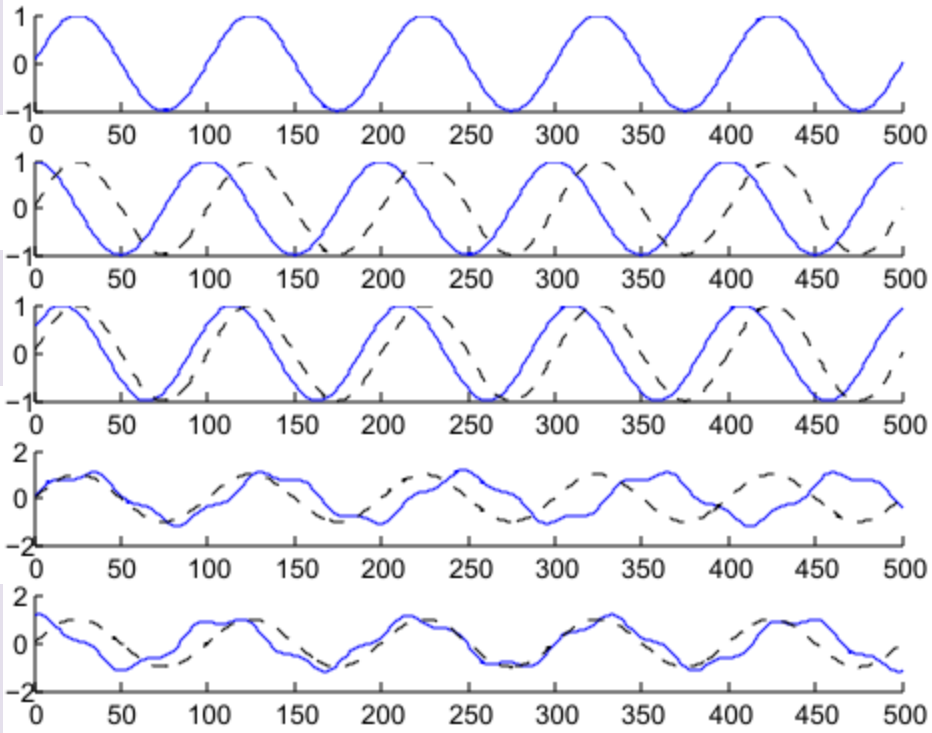
Why PCA fails?

- Properties of Good features / “fingerprints”:
  (a) Time shift / lag independent
  (b) frequency proximity
  (c) grouping harmonics

Example: synthetic signals

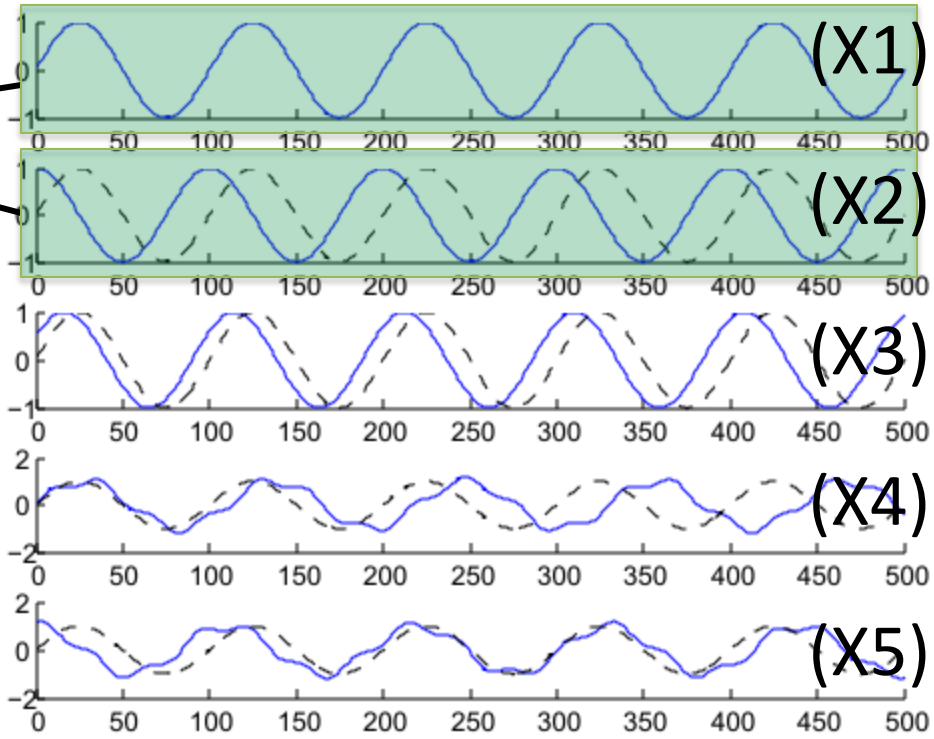
Equations

(X1)	$\sin(2\pi t/100)$
(X2)	$\cos(2\pi t/100)$
(X3)	$\sin(2\pi t/98 + \pi/6)$
(X4)	$\sin(2\pi t/110) + 0.2\sin(2\pi t/30)$
(X5)	$\cos(2\pi t/110) + 0.2\sin(2\pi t/30 + \pi/4)$



Intuition of “fingerprints”

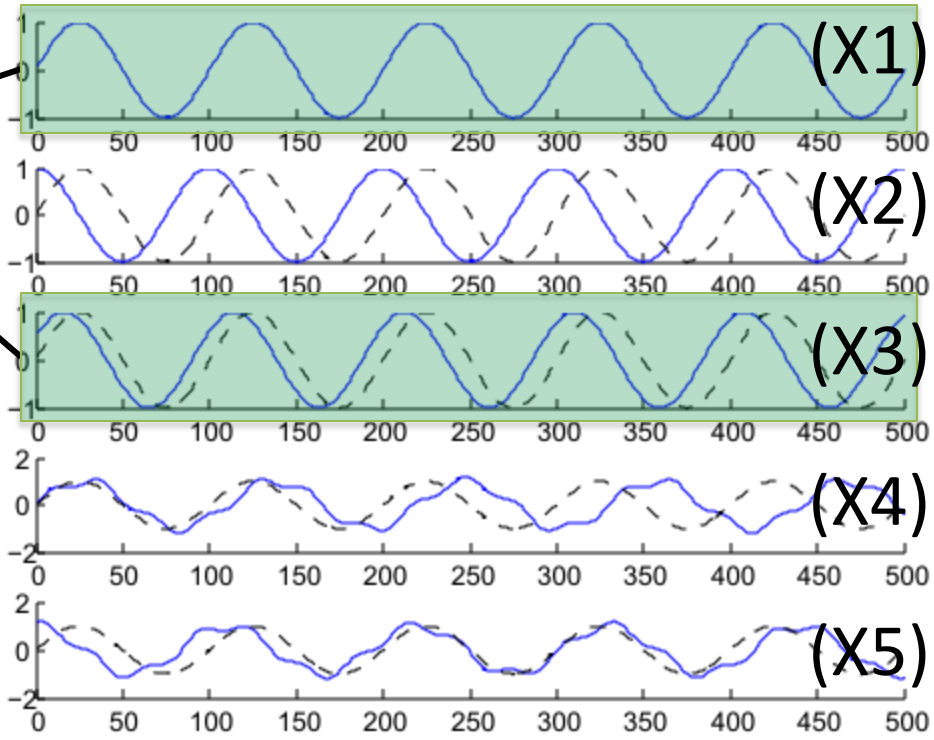
(a) Time shift



e.g.
left-foot-start walking
v.s.
right-foot-start walking

Intuition of “fingerprints”

(b)nearby frequency

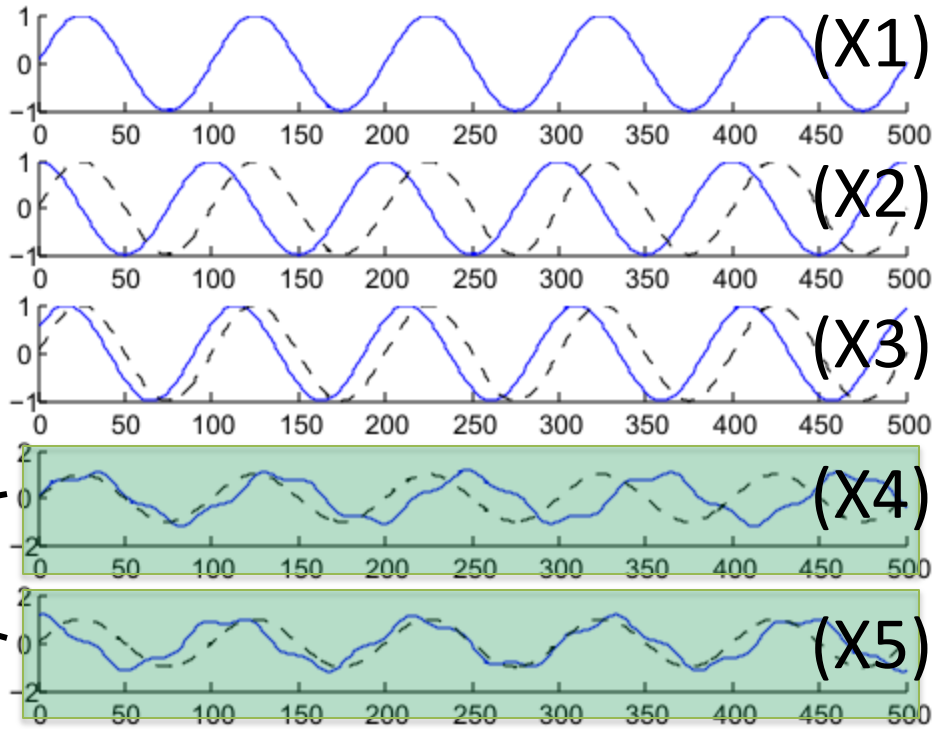


e.g.
running
v.s.
fast running

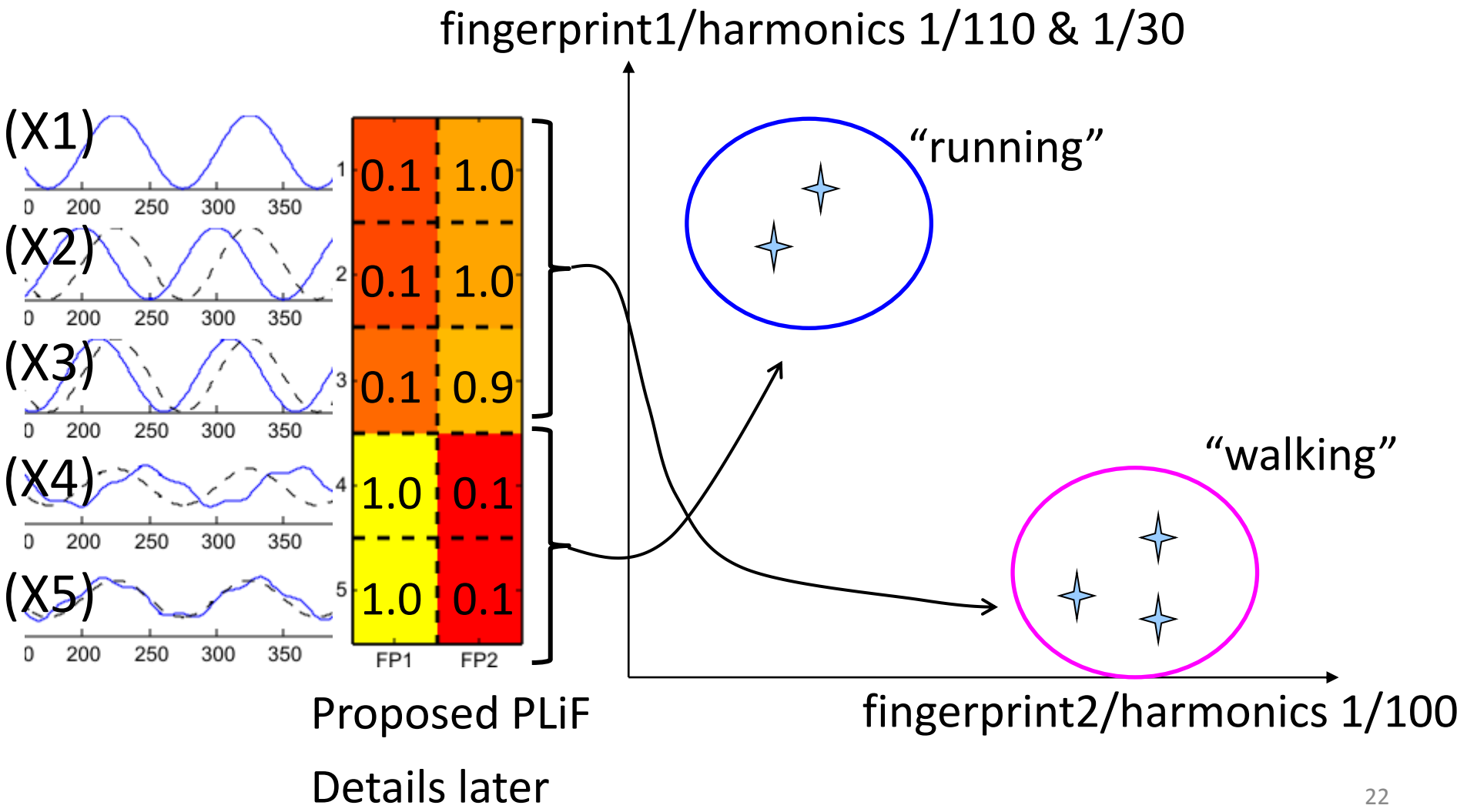
Intuition of “fingerprints”

~ human voices

(c) groups of harmonics



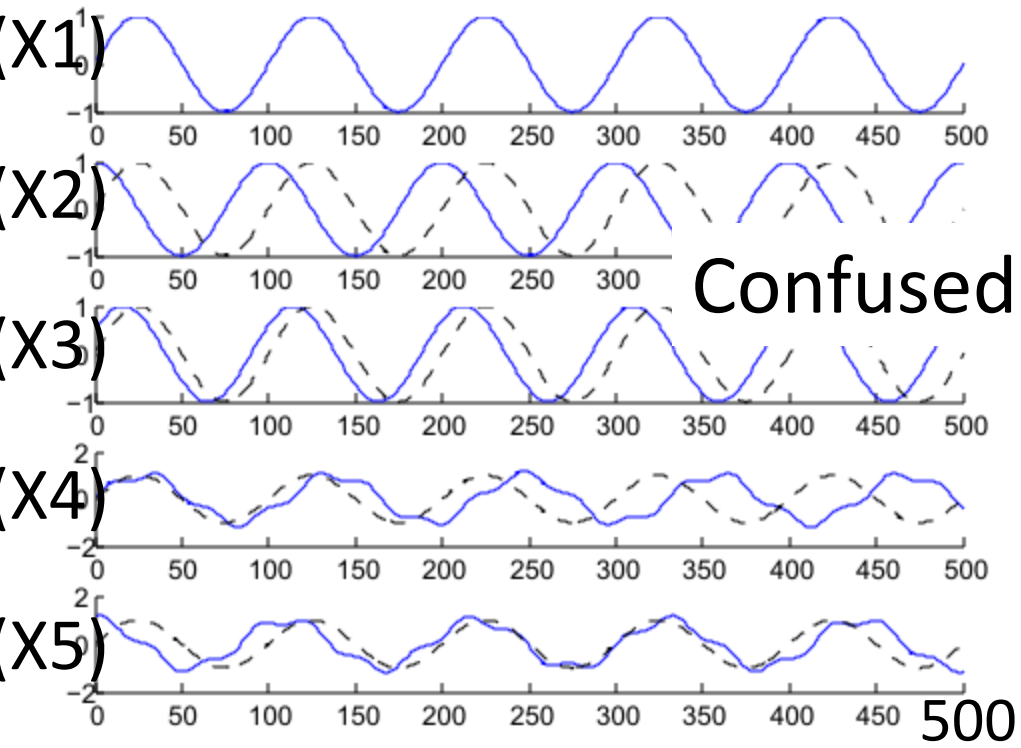
How to extract fingerprints?

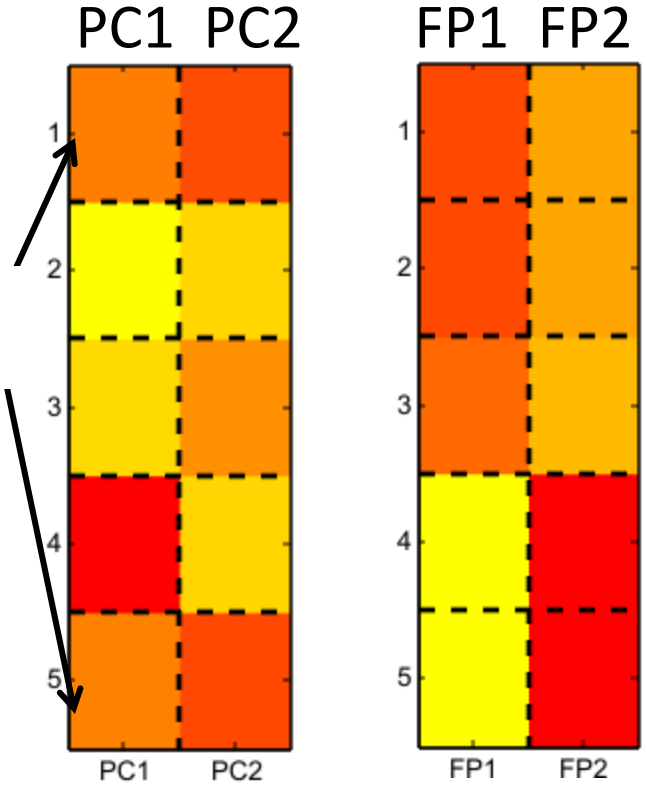


Proposed PLiF

Details later

For expert: Why not SVD/PCA?





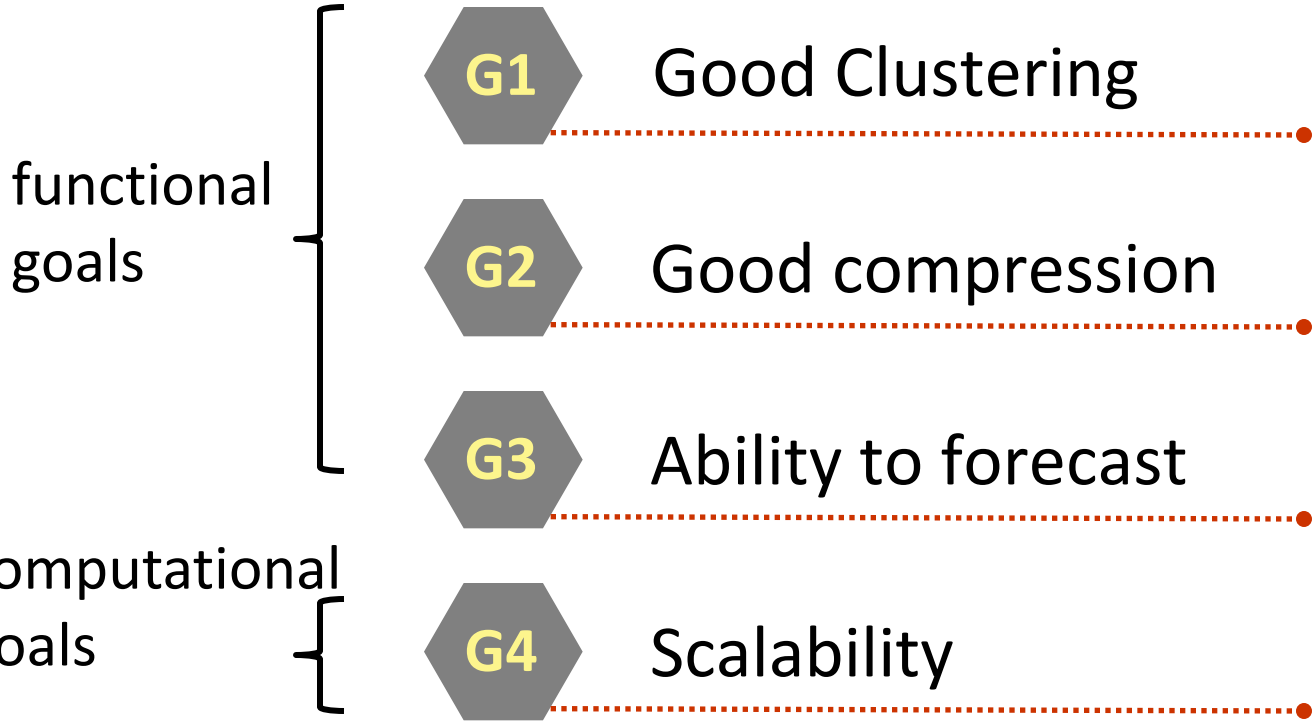
PCA
no clear grouping

PLiF

Beyond features

requirements
(a) lag independent
(b) frequency proximity
(c) grouping harmonics

Find good/interpretable fingerprints for

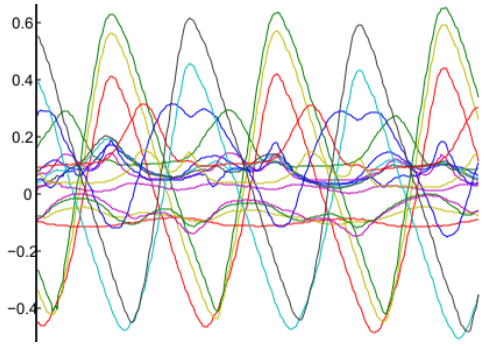


Outline

- Motivation
- An Obvious Solution & Related Work
- Proposed Method: Intuition & Example
- Experiments & Results
- PLiF: High Level Ideas
- PLiF: Low Level Details
- Conclusion

Datasets

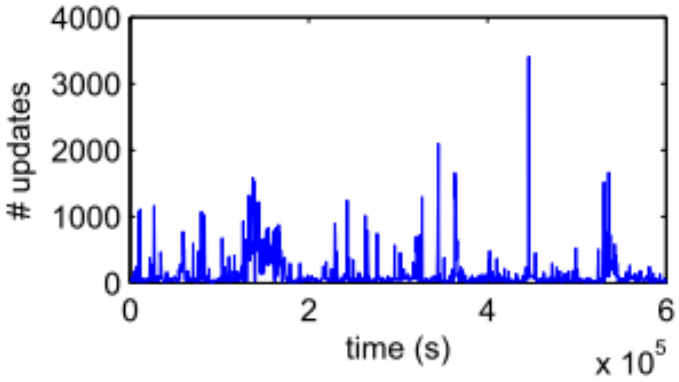
Mocap



Size: 49 × 100-500

mocap.cs.cmu.edu

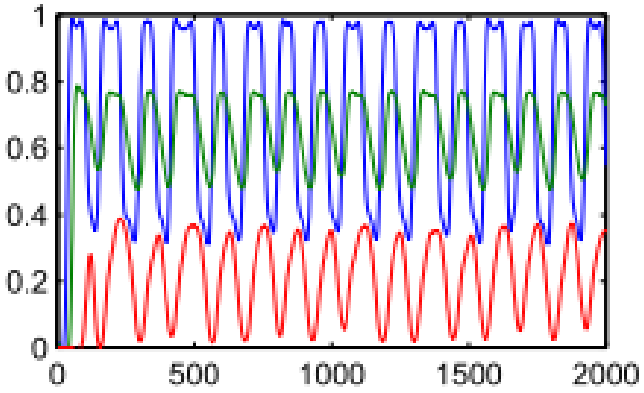
BGP network traffic



Size:10 × 103k

www.datapository.net

Chlorine



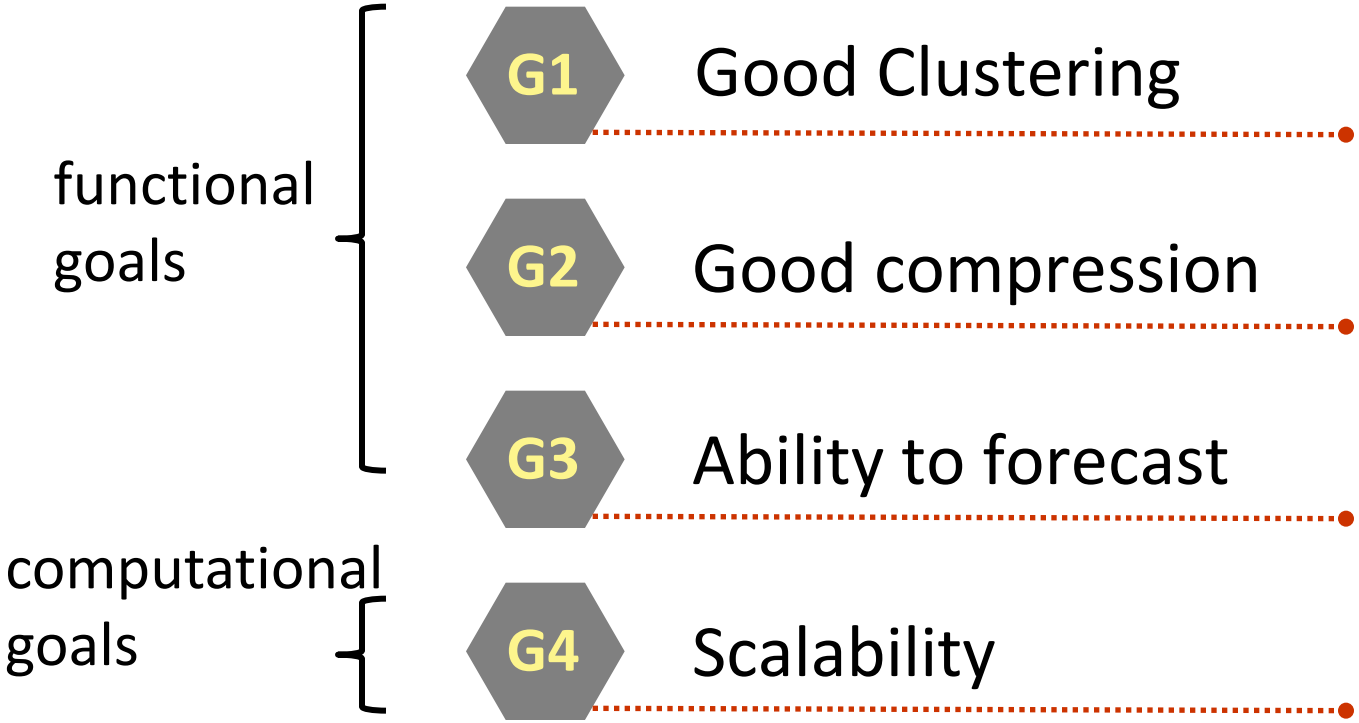
Size:166 × 4k

courtesy of

Prof. J. M. VanBriesen

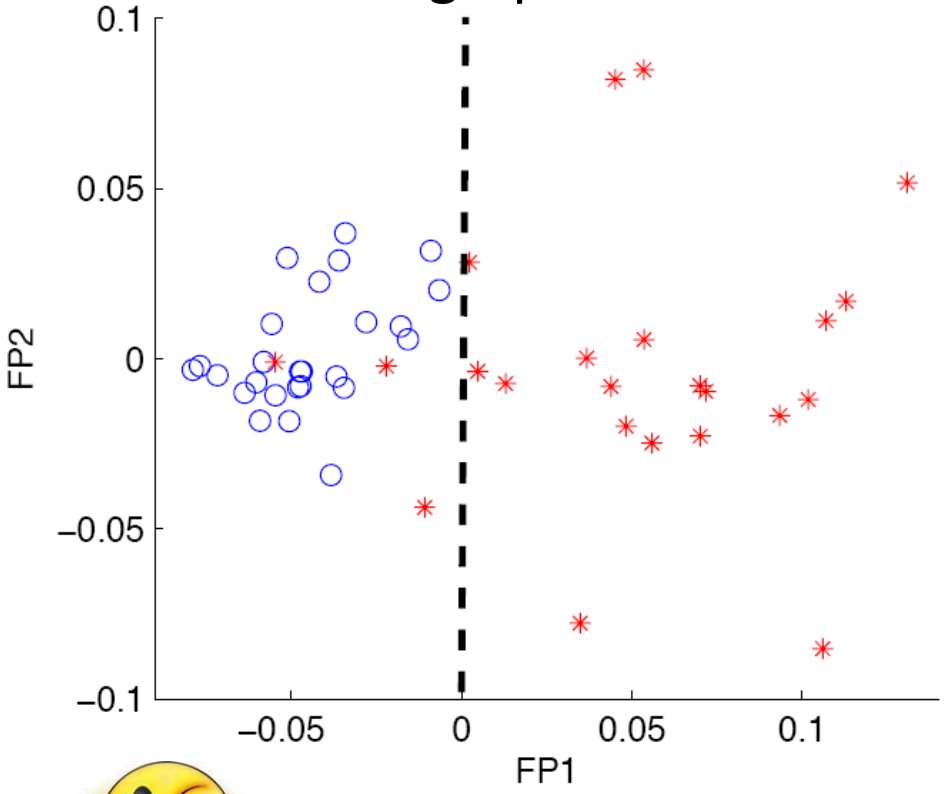
Experiment: Goals to meet

Find good/interpretable fingerprints for



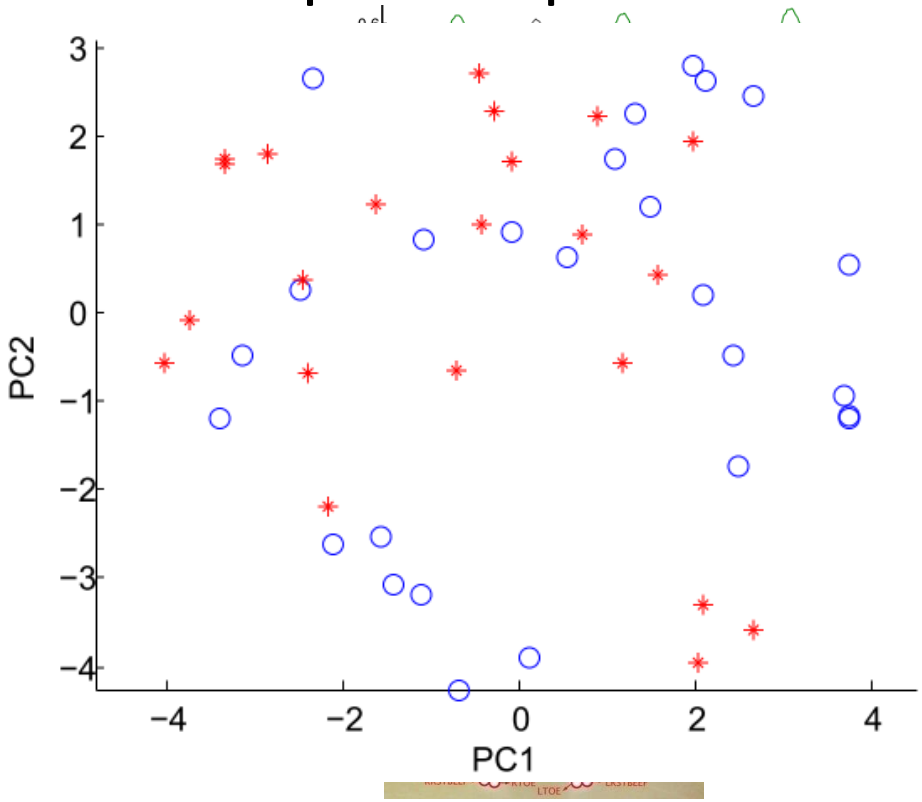
Result – Clustering

PLiF two “fingerprints”



Accuracy = 93.9%

PCA top 2 components

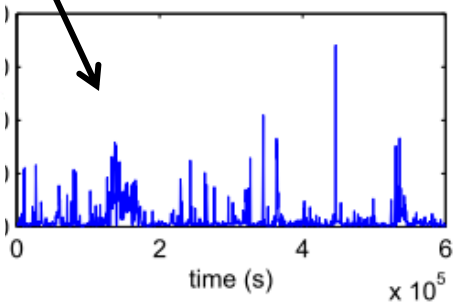


Accuracy = 51.0%

○ walking motion ✳ running motion

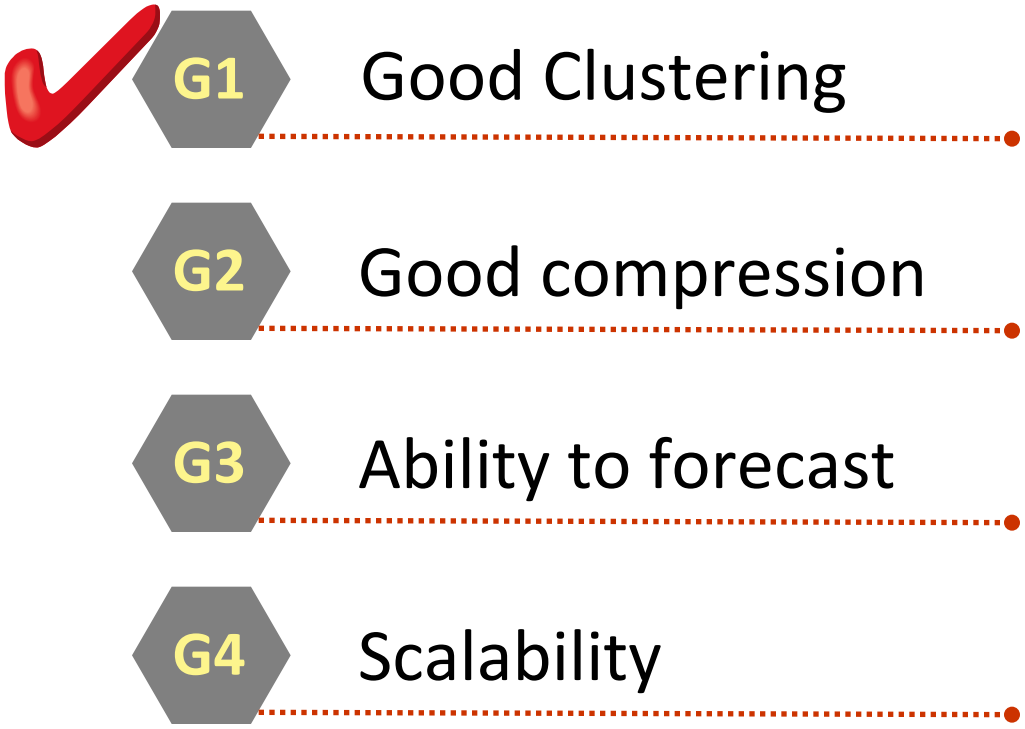
Result – Clustering

BGP data: PLiF + hierarchical clustering



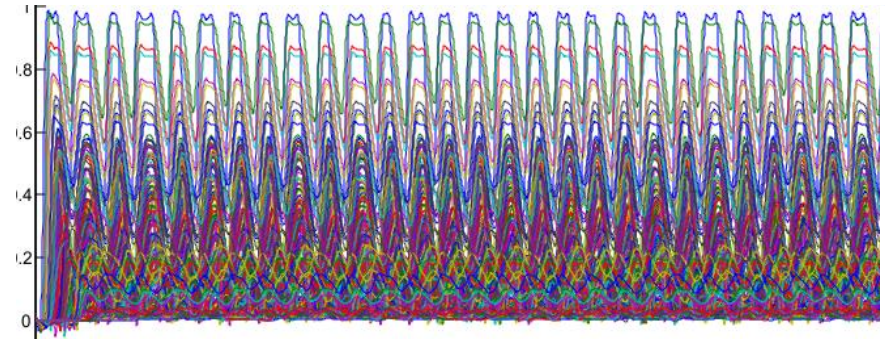
Experiment: Goals

Find good/interpretable fingerprints for

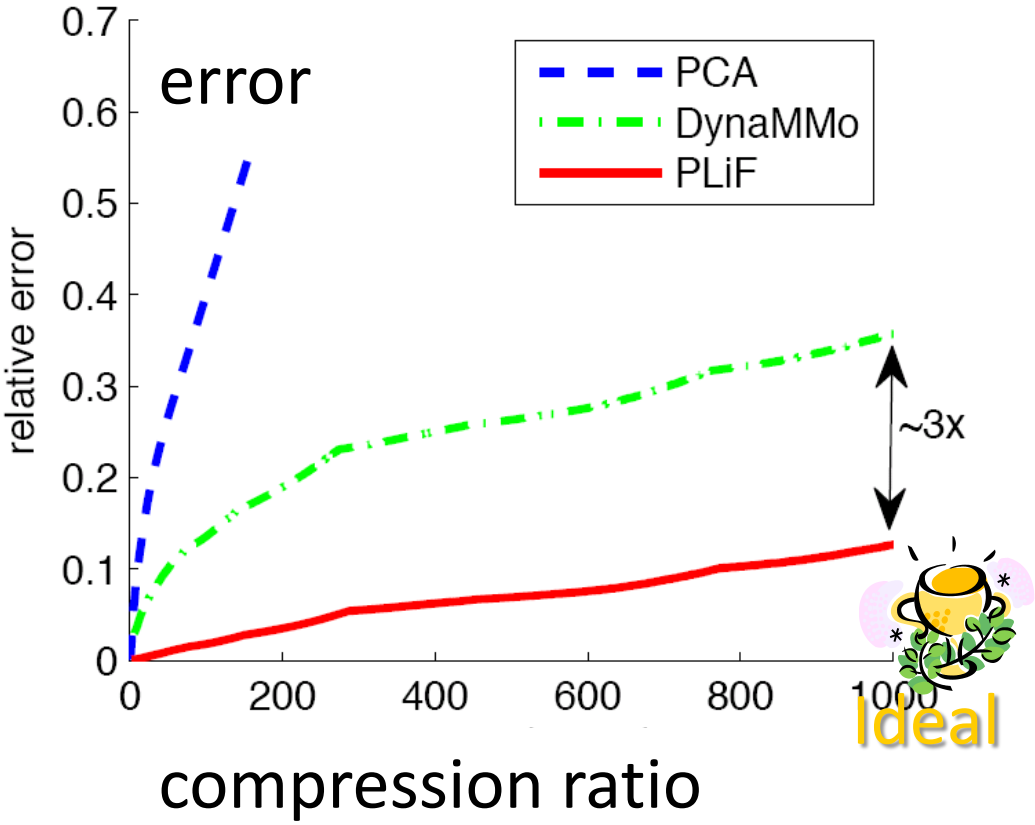


Result - Compression

Chlorine 166 * 4k

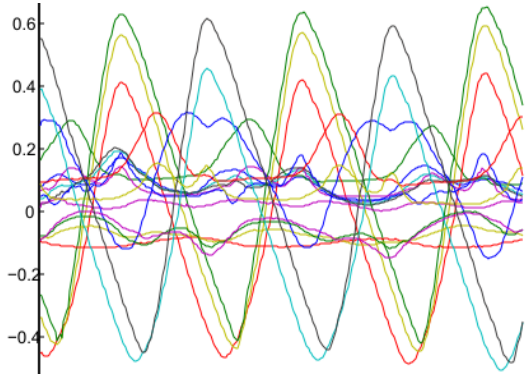


Storing only the PLiF features
& sampling of hidden variables



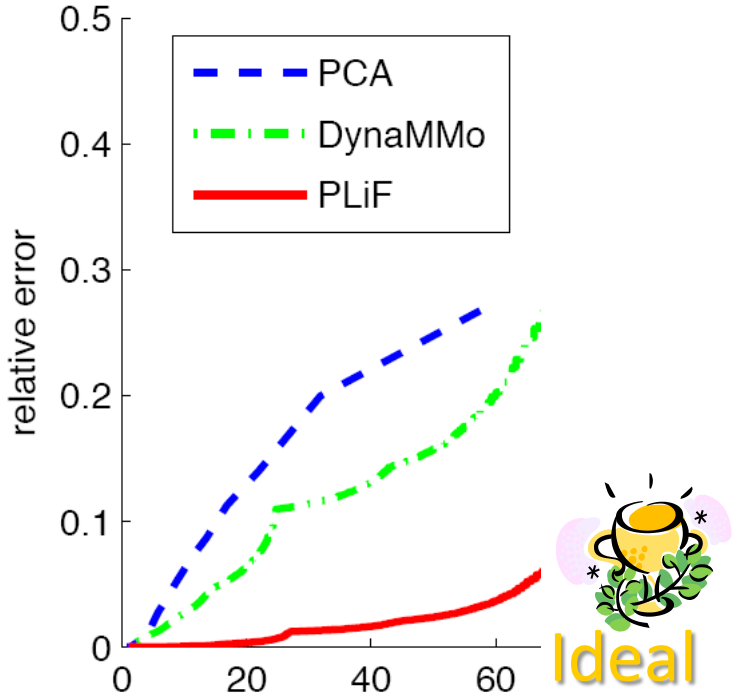
Result - Compression

Mocap: 93 * 300



Storing only the PLiF features
& sampling of hidden variables

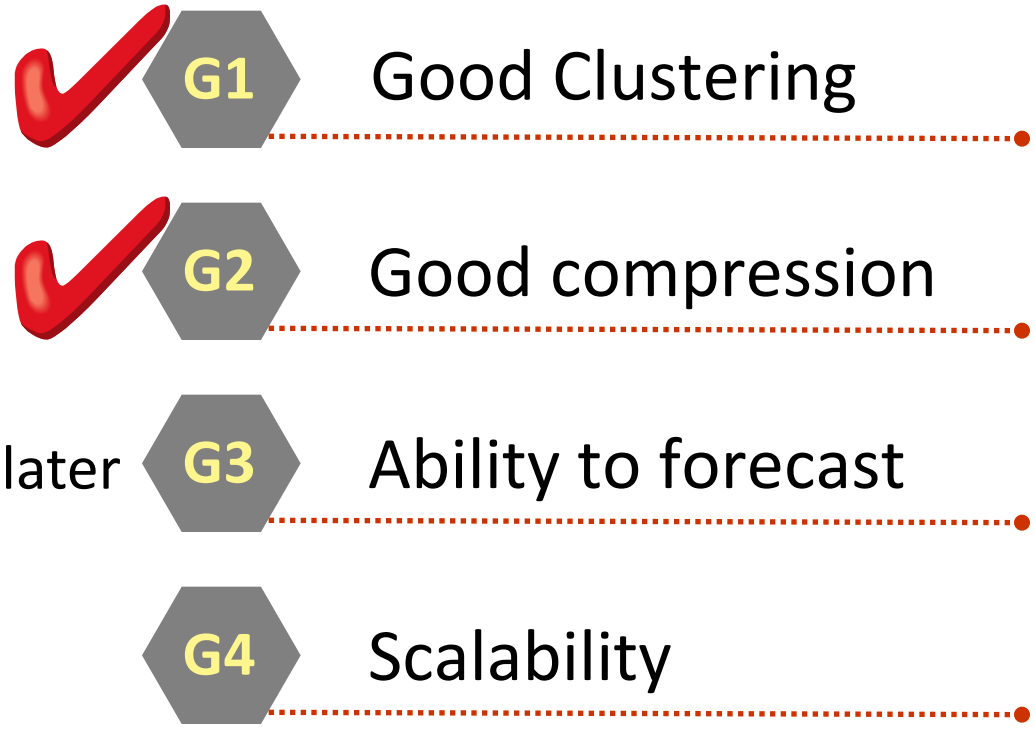
error



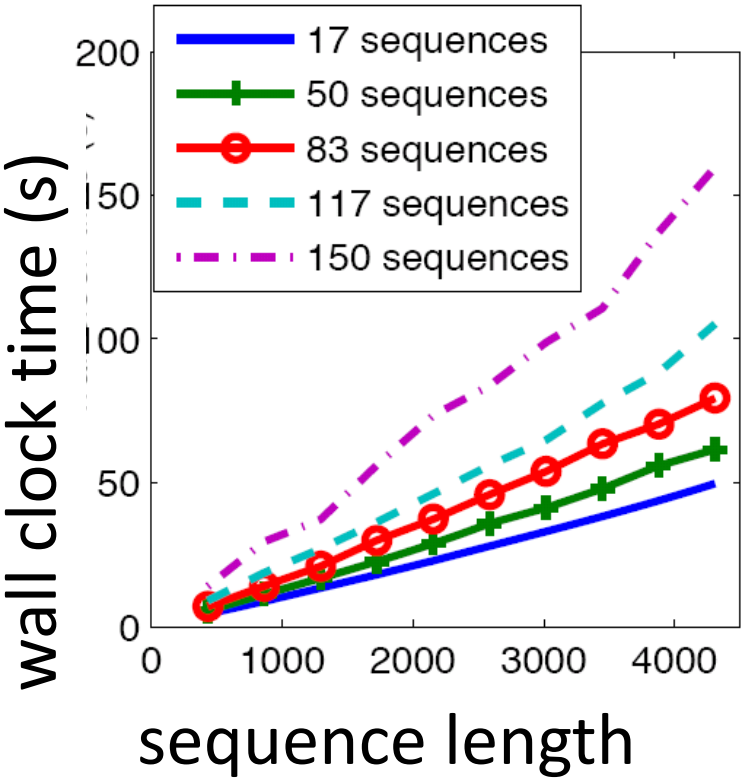
compression ratio

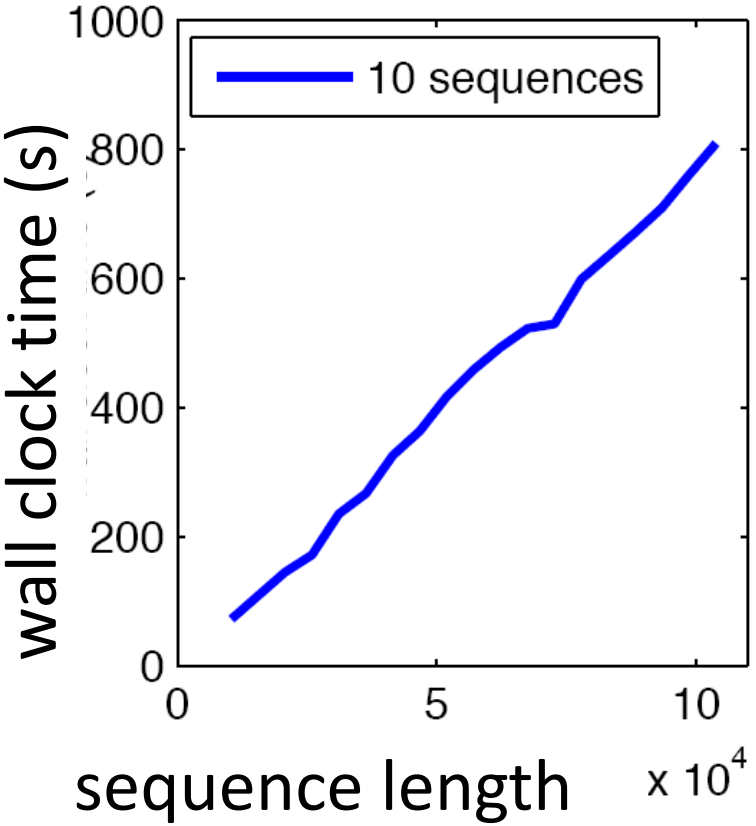
Experiment: Goals

Find good/interpretable fingerprints for



Scalability





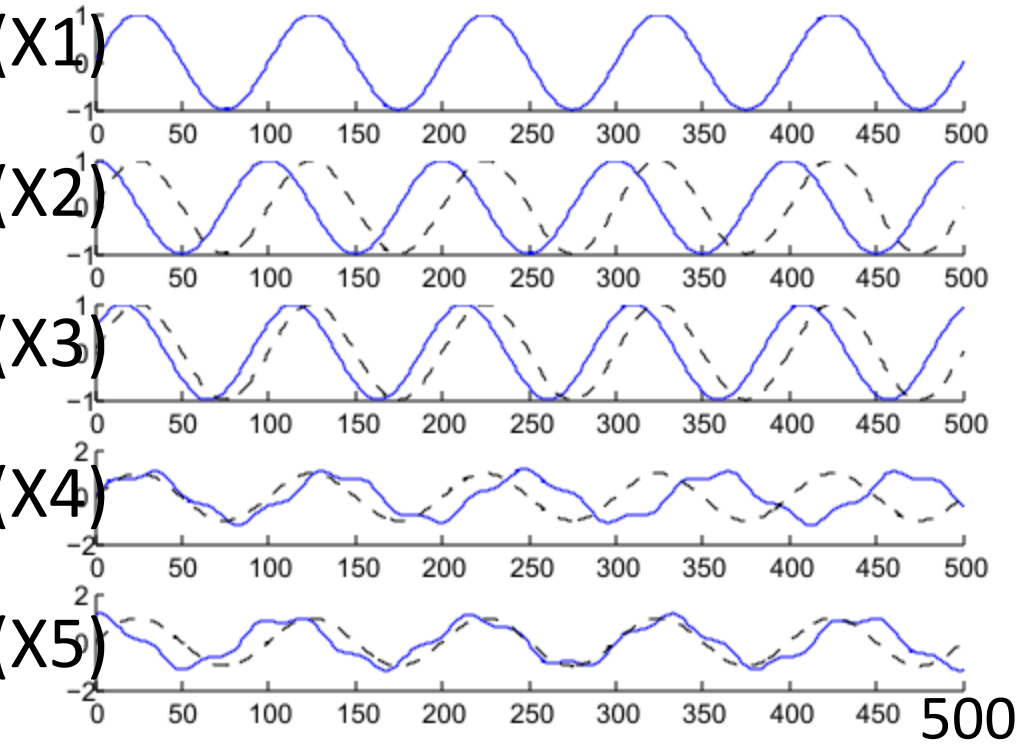
Linear to sequence length!

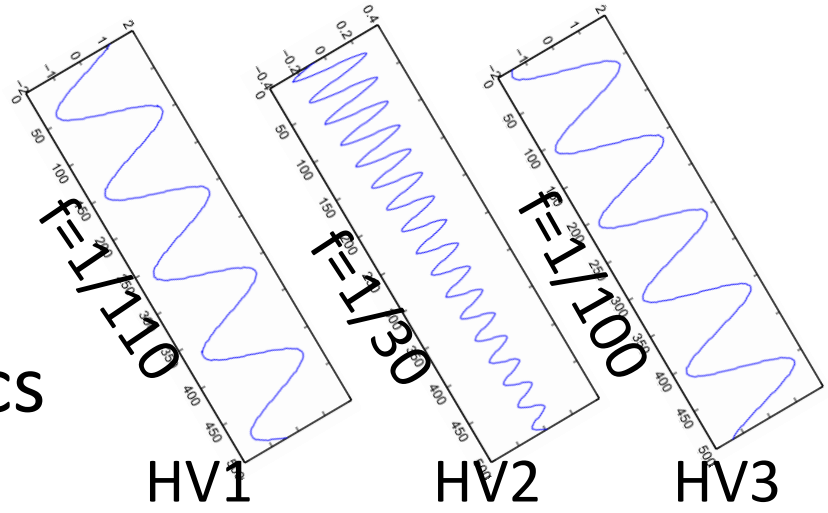
Outline

- Motivation
- An Obvious Solution & Related Work
- Proposed Method: Intuition & Example
- Experiments & Results
- PLiF: High Level Ideas
- PLiF: Low Level Details
- Conclusion

how PLiF works?

find hidden variable/harmonics

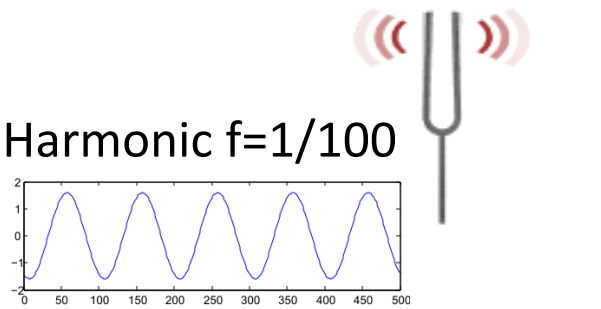


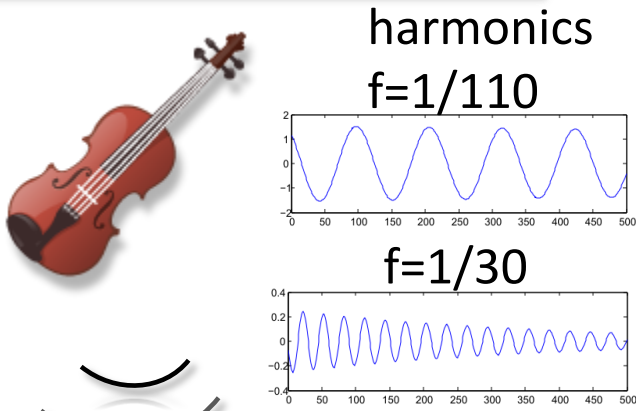


	HV1		HV2		HV3
=	0	+	1.0	+	0
=	0	+	1.0	+	0
=	0	+	0.9	+	0
=	1.0	+	0	+	1.0
=	1.0	+	0	+	1.0

Mixing weights

An Analog of Hidden Variables

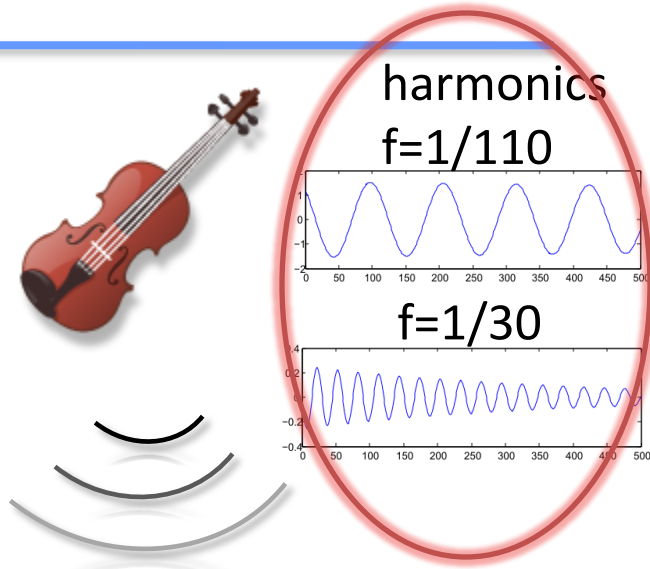


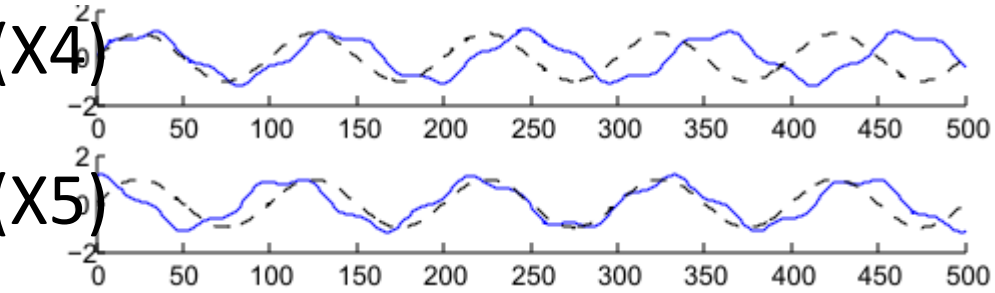


Mixing weights = participation strength of sound sources in observation (mic.)

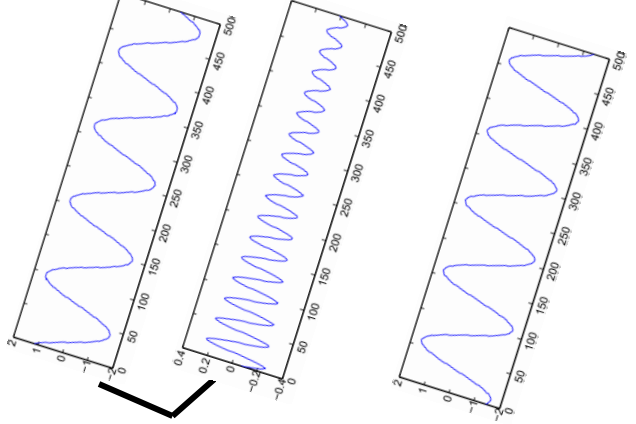
Grouping Correlated Harmonics

HV1’ = {HV1, HV2}

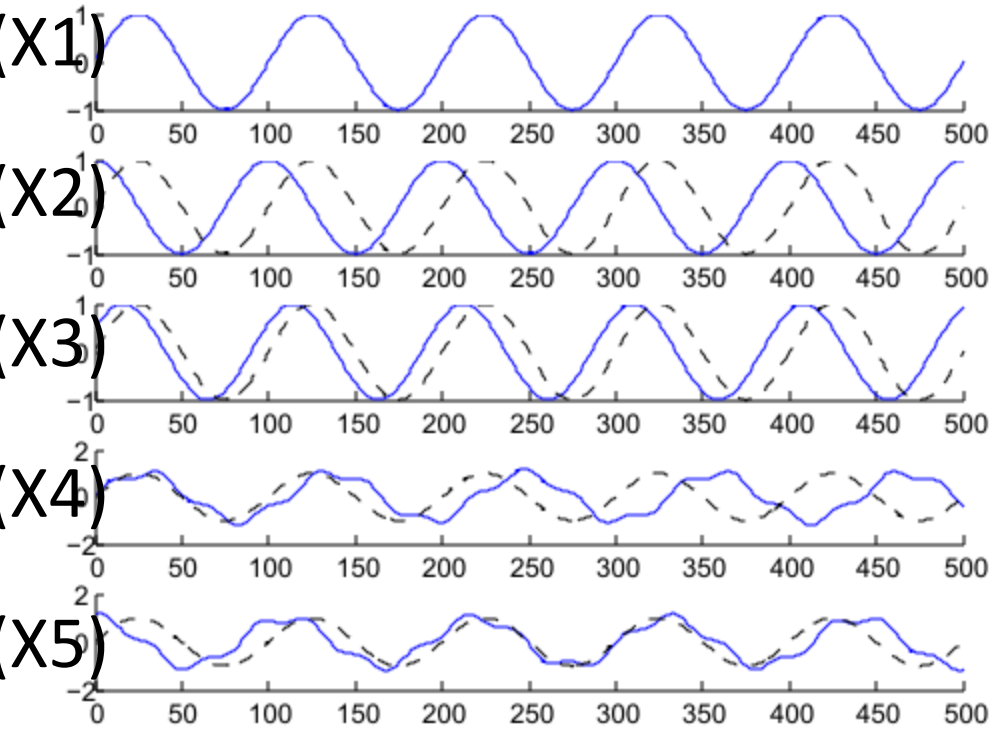




Fingerprints



HV1’ HV2’(=HV3)



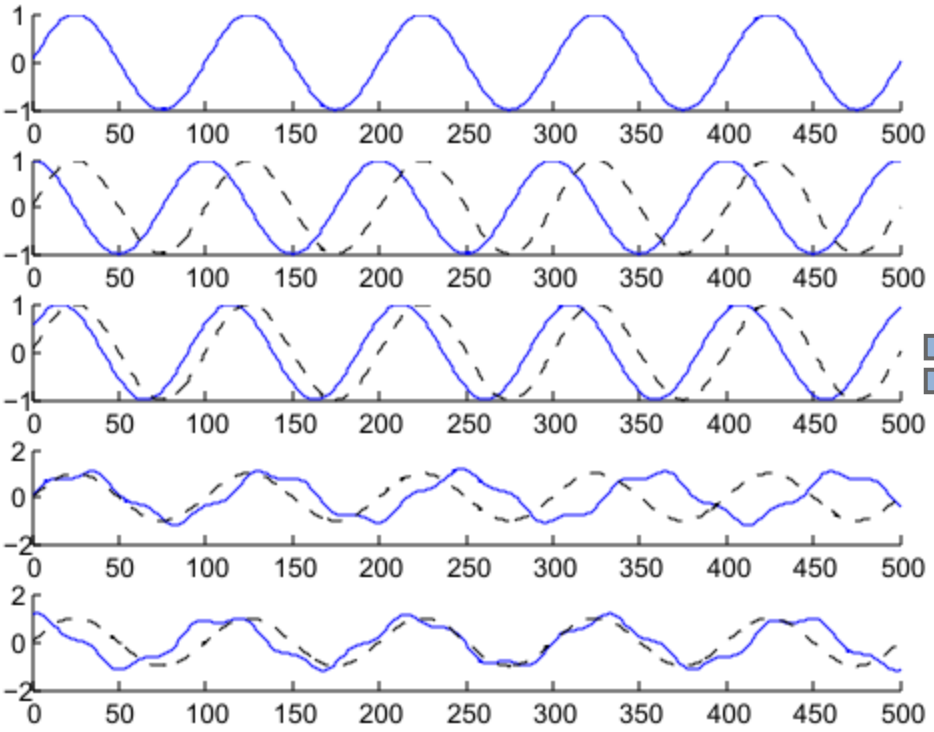
= 0 + 1.0

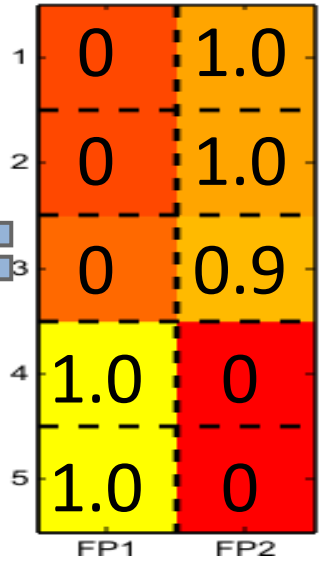
= 0 + 1.0

= 0 + 0.9

= 1.0 + 0

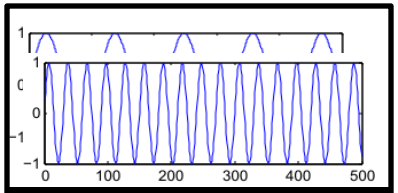
= 1.0 + 0



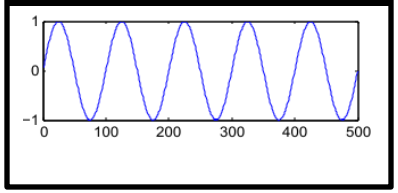


“ ”

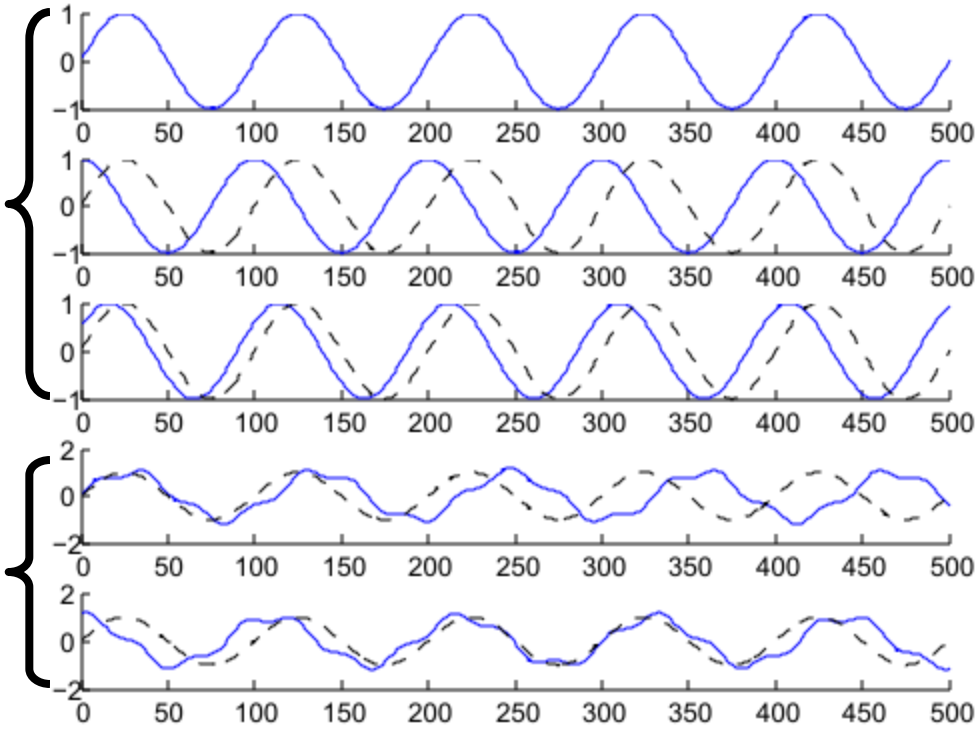
HV1’



HV2’

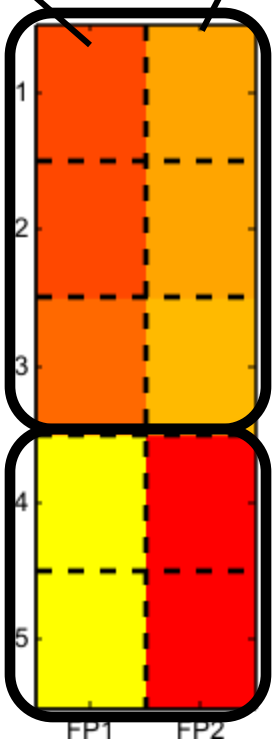


How to interpret?



Group of harmonics 1/110 & 1/30

harmonic. 1/100



Proposed PLiF

Outline

- Motivation
- An Obvious Solution & Related Work
- Proposed Method: Intuition & Example
- Experiments & Results
- PLiF: High Level Ideas
- PLiF: Low Level Details
- Conclusion

Steps of PLiFL: Overview

Learning Dynamics

Kalman/LDS

Finding Canonical Form

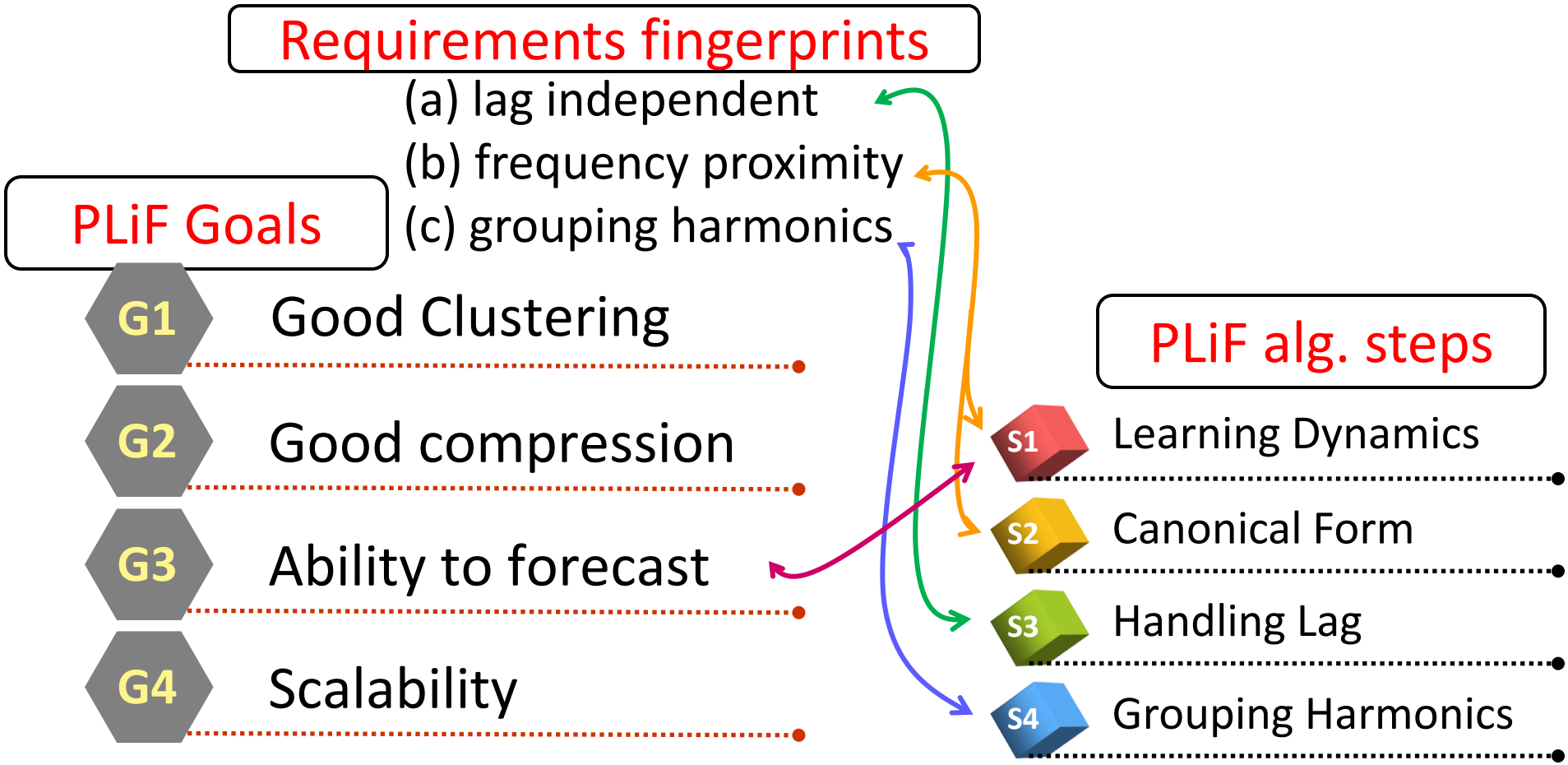
Eigenvalue decomposition

Handling the Lag

Grouping Harmonics

SVD

Why to do …?


Requirements fingerprints
(a) lag independent
(b) frequency proximity
(c) grouping harmonics
PLiF Goals
G1
Good Clustering
G2
Good compression
G3
Ability to forecast
G4
Scalability
PLiF alg. steps
S1
Learning Dynamics
S2
Canonical Form
S3
Handling Lag
S4
Grouping Harmonics

Warning!
A lot math
Only if you want to implement

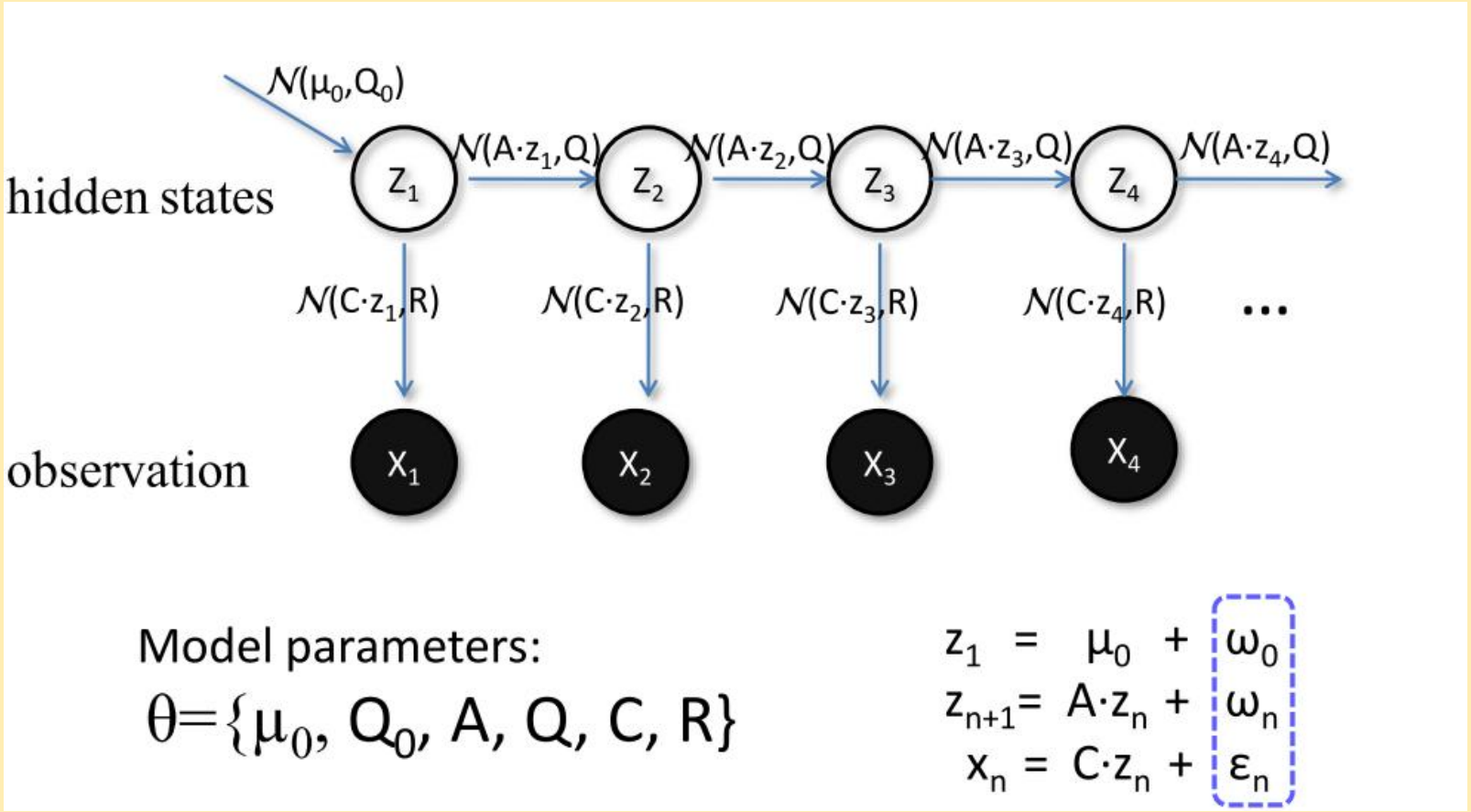
http://www.cs.cmu.edu/~leili/

Step 1. Learning Dynamics

- Use Kalman filters / Linear Dynamical Systems (LDS) to learn the hidden variables

Details

Underlying Model: Linear Dynamical Systems



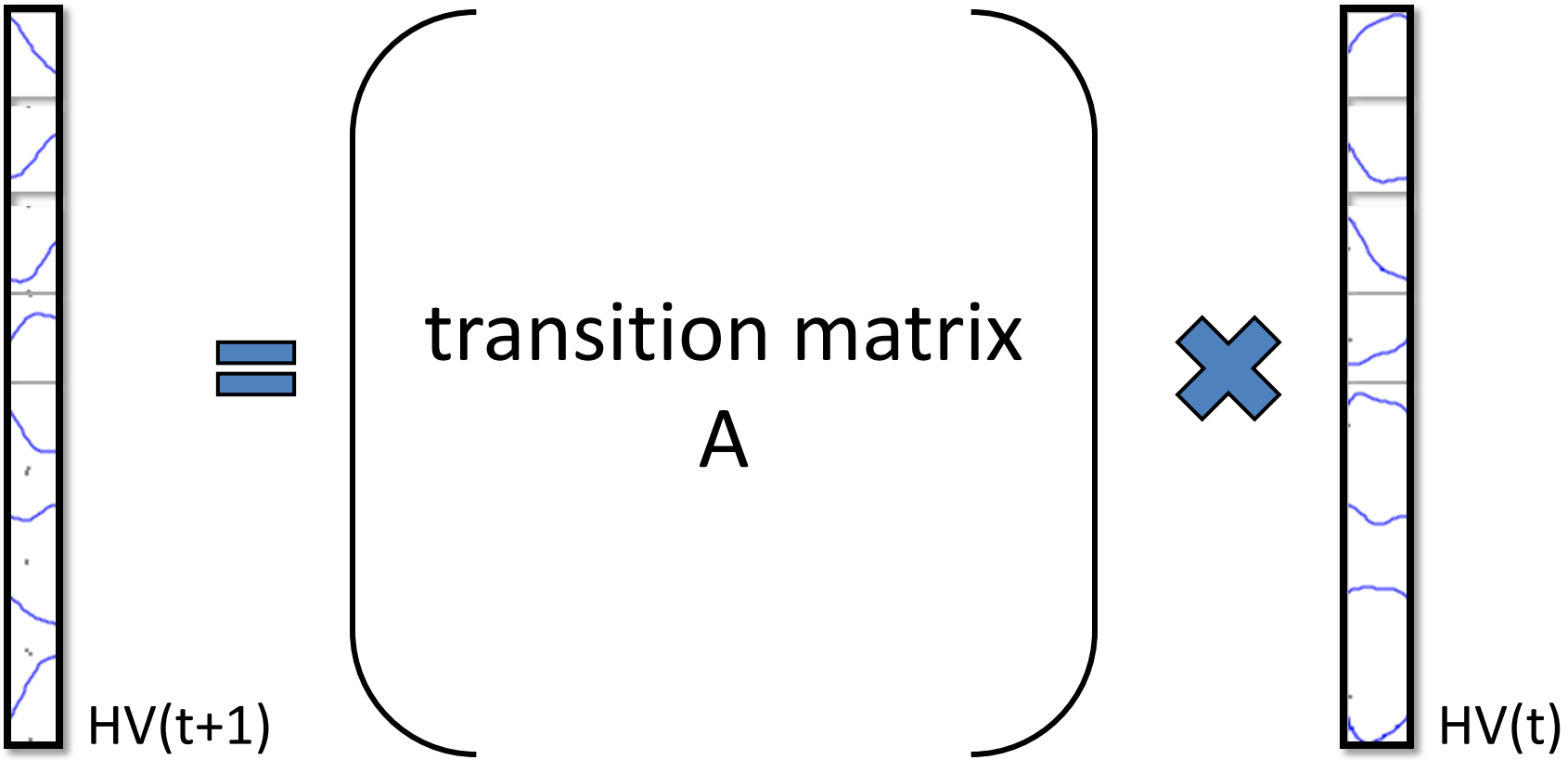
Model parameters:

$\theta = \{\mu_0, Q_0, A, Q, C, R\}$

$$z_1 = \mu_0 + \omega_0$$
$$z_{n+1} = A \cdot z_n + \omega_n$$
$$x_n = C \cdot z_n + \varepsilon_n$$

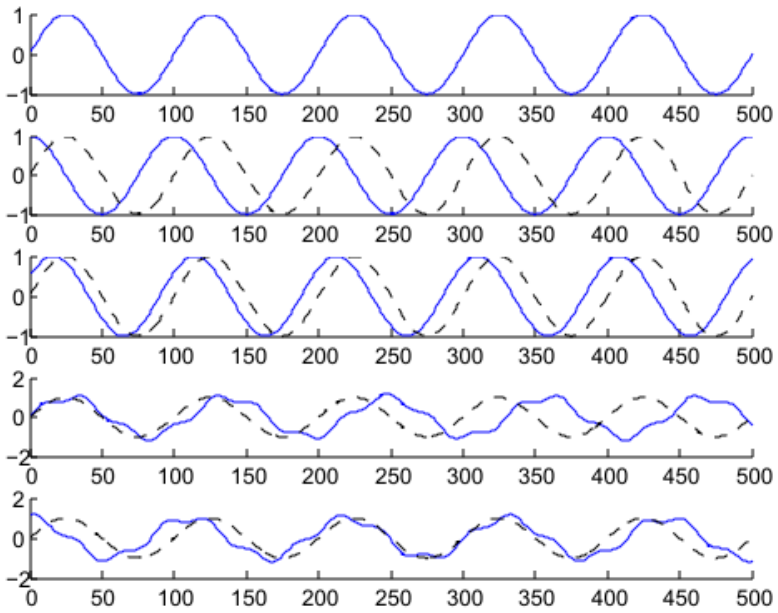
Dynamics/Transition in Hidden Variables

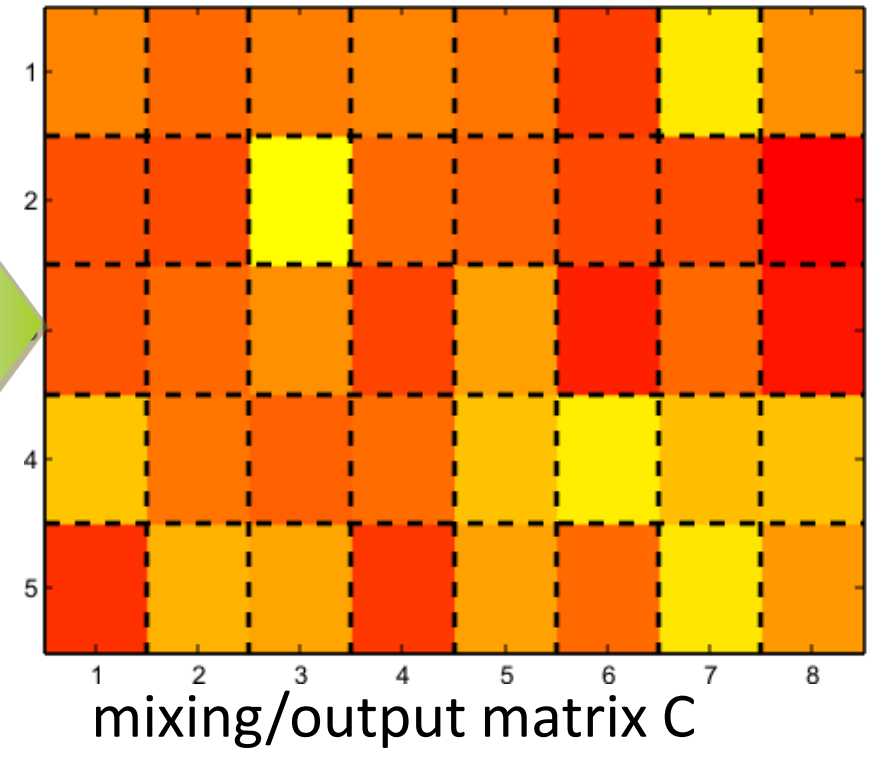
- enables forecasting



Learning Mixing Weights

Expectation-Maximization algorithm [Ghahramani 96]





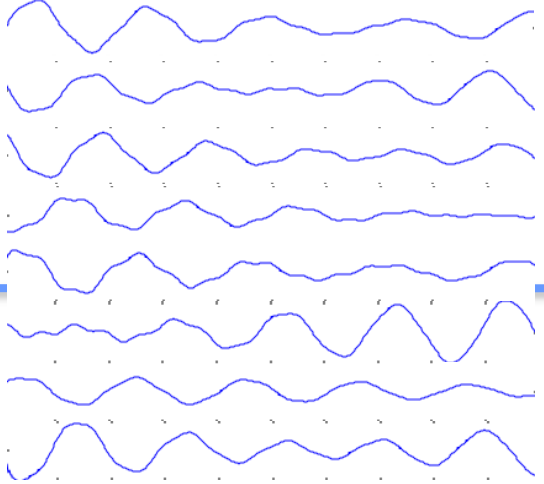
mixing/output matrix C

Step 2: Canonicalization

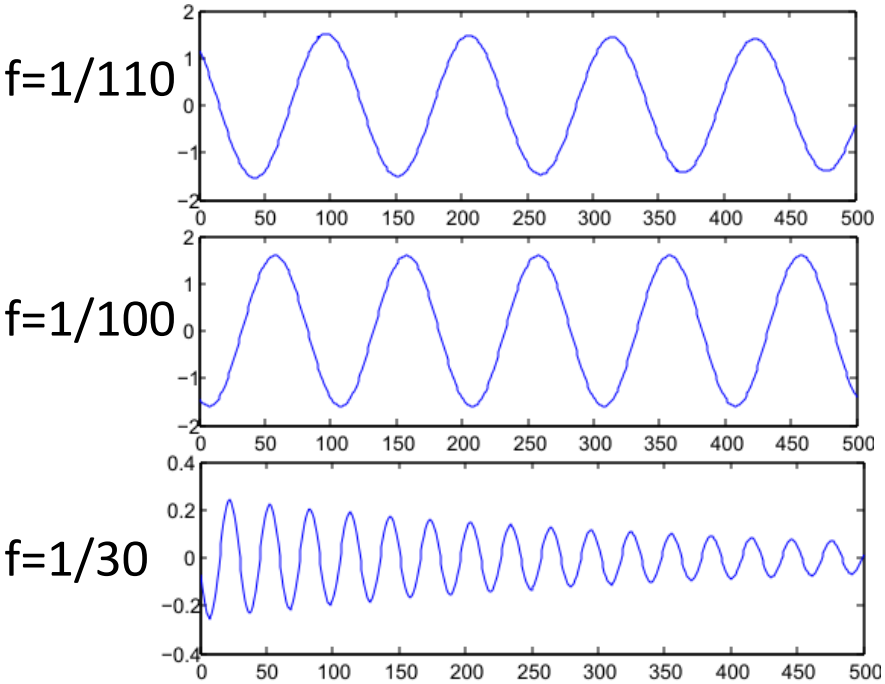
- Use eigen-decomposition on transition matrix A to find “harmonics” and mixing weight of harmonics

$$\texttt{find}\ \ \Lambda\ V$$
$$A \cdot V = \Lambda \cdot V$$

Canonicalization adds Interpretability



HV before

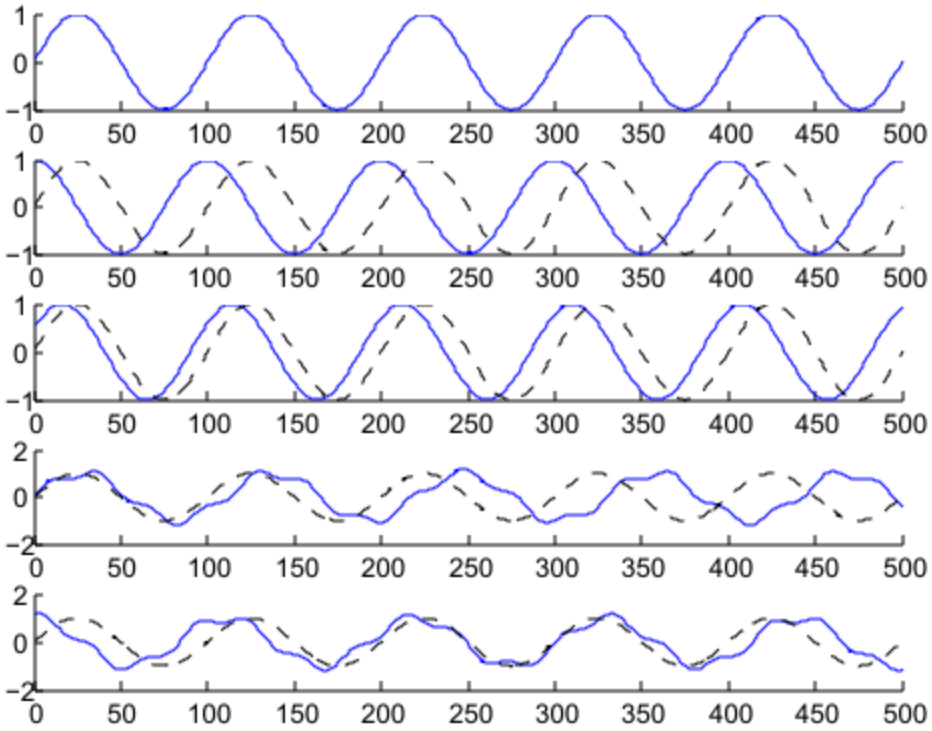


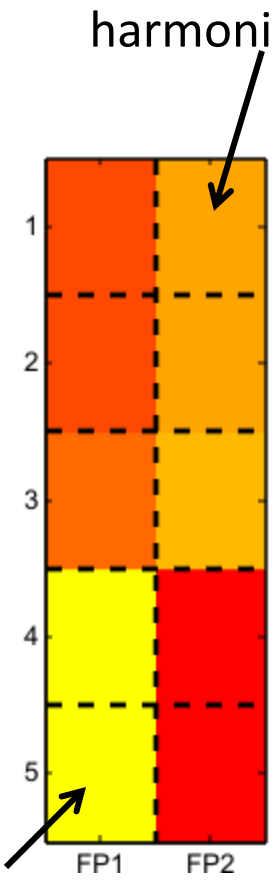
"Harmonics" → frequency

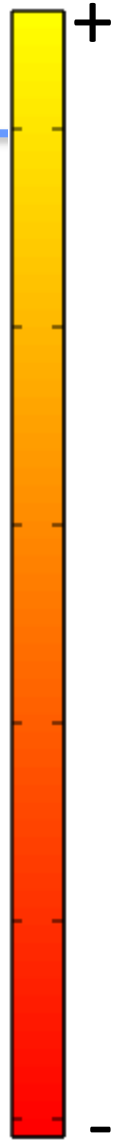
"Harmonics" → Growing/ shrinking trend

Time series of HV after canonicalization (real part)

Step 4: Grouping Harmonics



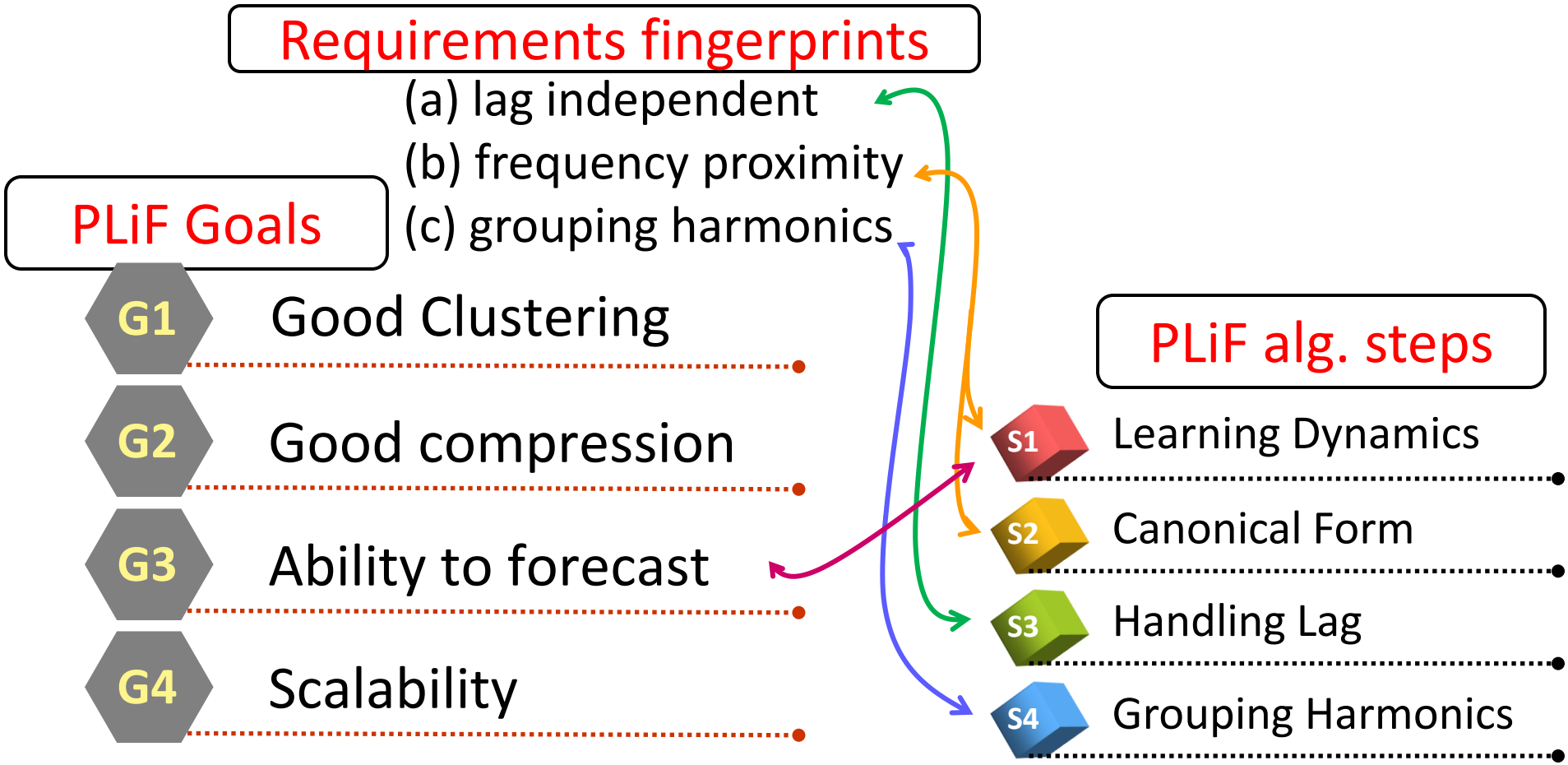




Conclusion

- Intuition of PLiF
  - three requirements of fingerprints
- How it works
  - Four steps in the algorithm
- What to do with PLiF
  - Similarity, clustering, compression, forecasting, etc.
- Experiments on a diverse set of data
  - It really works
  - It is fast & scalable

Overview


Requirements fingerprints
(a) lag independent
(b) frequency proximity
(c) grouping harmonics
PLiF Goals
G1
Good Clustering
G2
Good compression
G3
Ability to forecast
G4
Scalability
PLiF alg. steps
S1
Learning Dynamics
S2
Canonical Form
S3
Handling Lag
S4
Grouping Harmonics

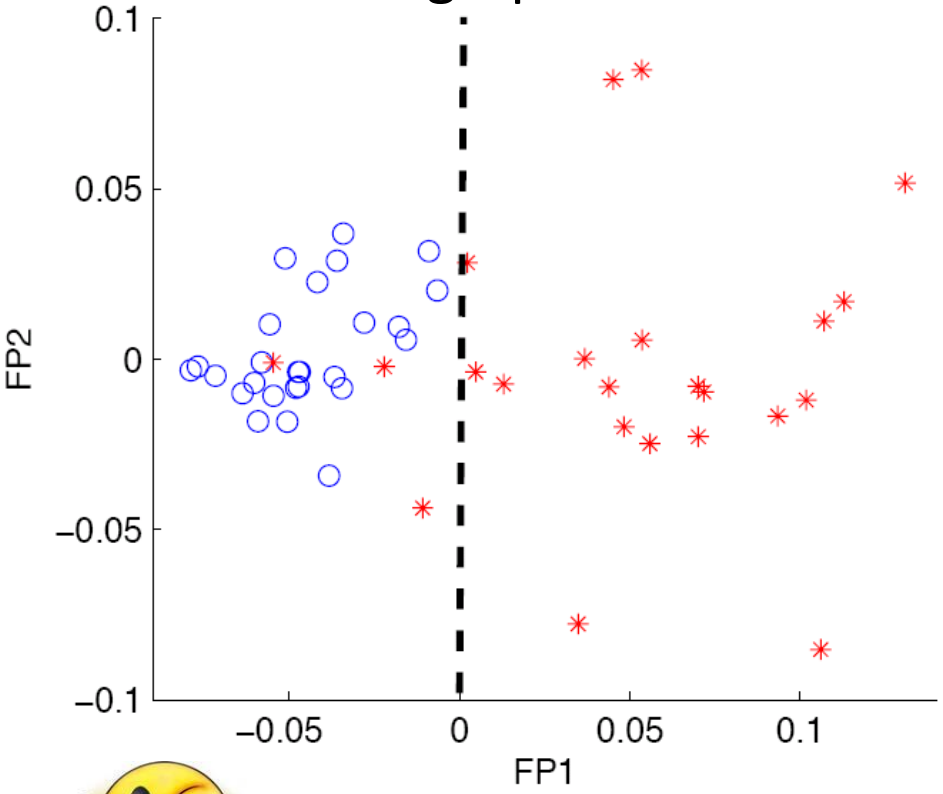
Future Work

- Non-Gaussian noise
- Nonlinear transition

Thanks!

www.cs.cmu.edu/~leili

PLiF two “fingerprints”



Accuracy = 93.9%

e

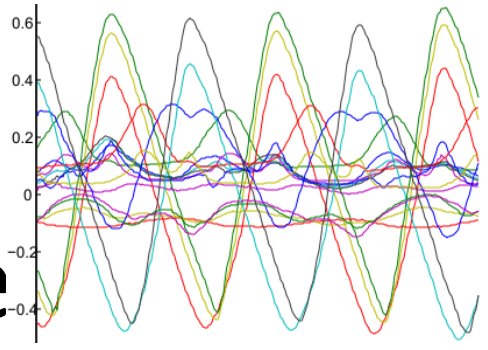
or Time

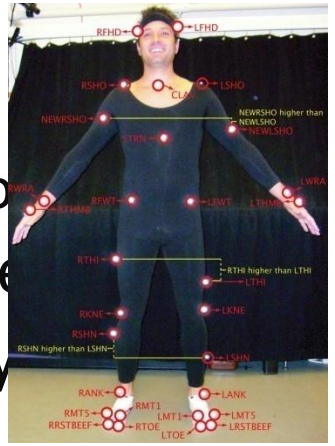
Li

akash, Christo

outer Science

on University





○ walking motion ✳ running motion