Multi-scale Geometric Summaries for Similarity-based Upstream Sensor Fusion

Christopher Tralie, Paul Bendich, John Harer

Duke University, ECE / Math

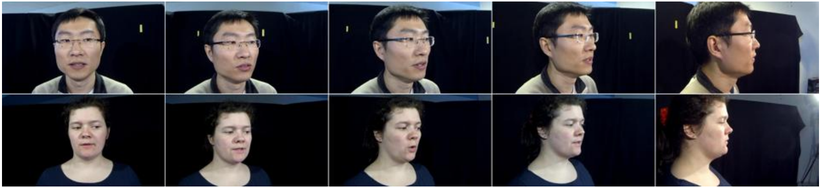
3/6/2019

Overall Goals / Design Choices

- ▷ Leverage multiple, heterogeneous modalities in identification
- ▷ Develop general tools without domain specific models
- ▷ Techniques are *unsupervised* (no training data required)

OuluVS2 Digits Dataset

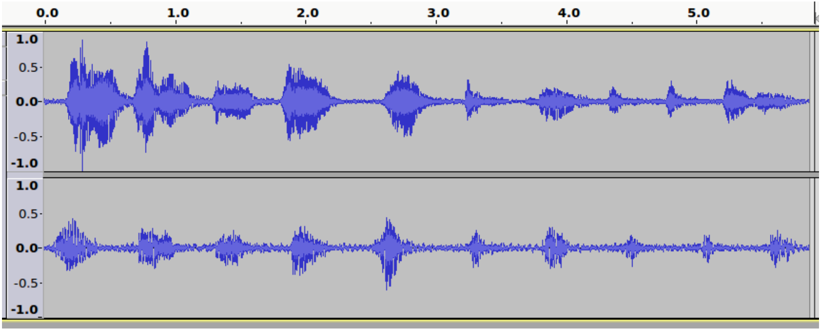
- 51 speakers
- 10 sequences, 3 instances per speaker per sequence
- Video from multiple points of view, audio



`http://www.ee.oulu.fi/research/imag/OuluVS2/index.html`

Why Digits?

- ▷ Modalities capture different aspects ("p" versus "b")
- ▷ Variation across speakers and across runs



- ▷ Even after uniformly scaling, the raw audio signals do not align perfectly in time

Problems And Success Metrics

- ▷ Decompose set of digit strings various ways:
  - ▶ by digit string, by speaker, by speaker and digit string
- ▷ Goal is to come up with similarity ranking mechanism μ s.t.
  - ▶ For each object s, $\mu(s, t)$ is larger when t is in same class as s

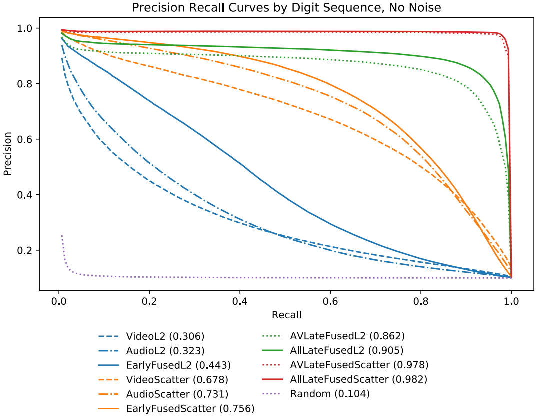
(Rusinkiewicz and Funkhouser 2009)

Problems And Success Metrics

- Success Evaluated by precision-recall curves for each object s
- *Recall*: Proportion of class items considered in an ordered list by similarity
- *Precision*: The proportion of items that are actually correct

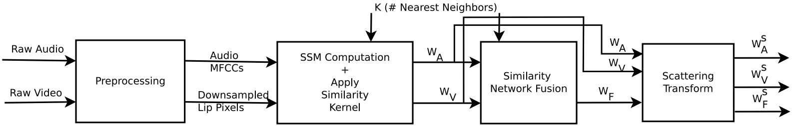
Problems And Success Metrics

- Success Evaluated by precision-recall curves for each object s
- Report average P-R curves
- Area under P-R curve is *mean average precision* (MAP)



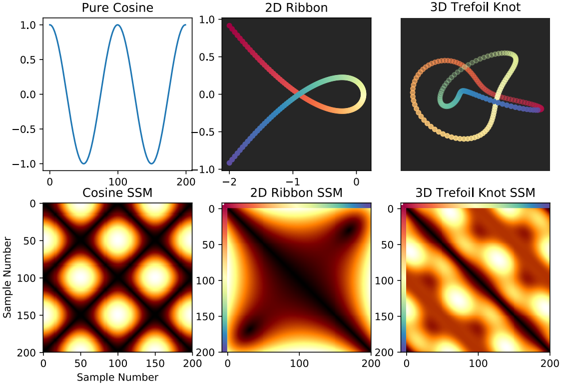
Other approches, our pipeline(s)

- Many approaches (including ours) construct μ via mapping strings into a feature space
- Lots of deep learning approaches (Lopez and Sukno, 2018)
- HMM per class, use canonical correlation analysis to learn good ways to extract fused audio/visual features (Sargin et al, 2007)
- We propose a set of entirely *unsupervised* pipelines
  - Labeled examples used only to *evaluate* not to *train*

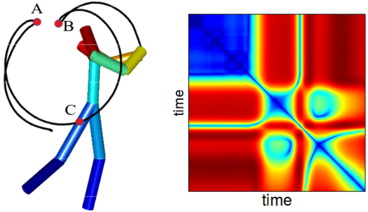


Self-Similarity Matrices (SSMs)

$$D_{ij} = ||X_i - X_j||_2$$



Why SSMs?



(a) (b) (c) (d)

Imran N Junejo et al. "View-independent action recognition from temporal self-similarities". In: *IEEE transactions on pattern analysis and machine intelligence* 33.1 (2011), pp. 172–185

SSMs on Our Data

Video:

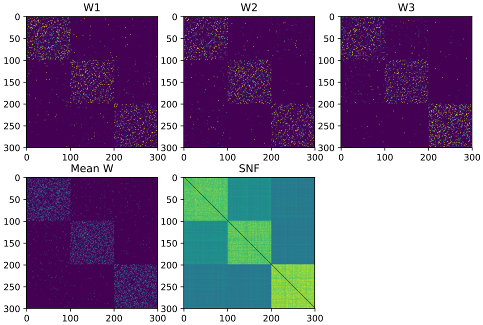
- Extract lip region from each frame and rescale to 25×25 grayscale
- Treat as time series in $25 \times 25 = 625$ dim Euclidean space

Audio:

- Break audio signal into overlapping windows
- Summarize each window via 20 MFCC coefficients
- Treat as time series in 20 dimensional Euclidean space

Similarity Network Fusion (SNF)

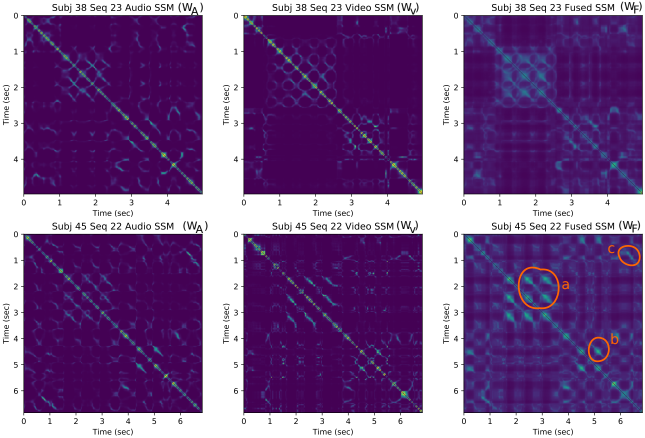
- Transform several weight matrices $W_1, \ldots, W_m$ into one that (hopefully) has best qualities of all
- Based on random walks with cross-talk between matrices for probabilities (works best if modalities are complementary)



Bo Wang et al. "Unsupervised metric fusion by cross diffusion". In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE. 2012, pp. 2997–3004

SNF for Early Audio-Visual Fusion

- We use SNF to fuse MFCC (audio) and lip pixel (video) SSMs

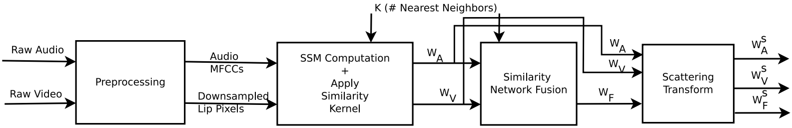


9 7 4 4 4 3 5 5 8 7

a: repeating 4s, b: repeating 5s, c: repeating 7s

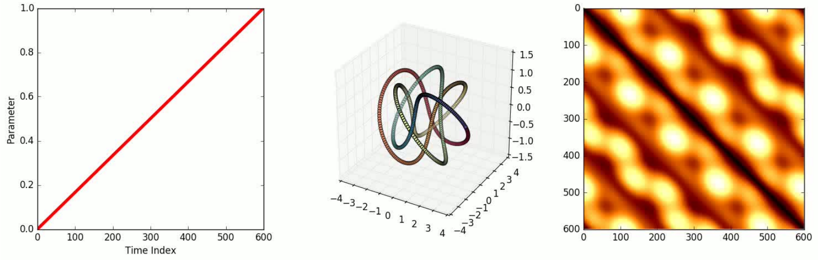
How To Compare (Fused) SSMs?

- Each string s transformed into SSM $W_A(s)$, $W_v(s)$, then fused into $W_F(s)$
- How to compare $W_F(s)$ with $W_F(s')$? Could just use ℓ_2 (Matrix Frobenius Norm)



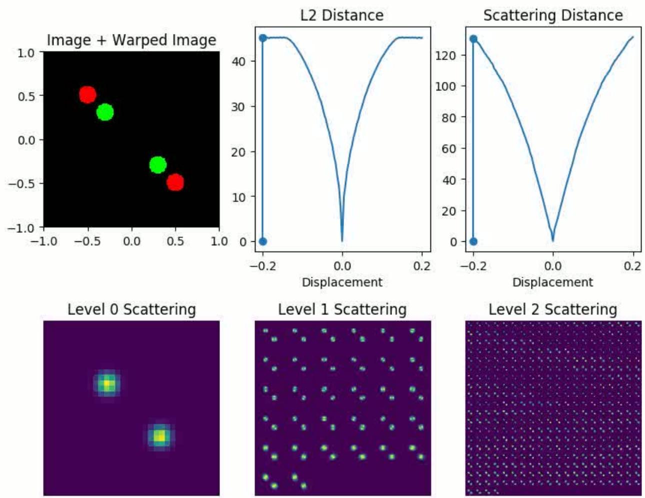
Measuring Similarity between SSMs

- Each string s transformed into SSM $W_A(s)$, $W_v(s)$, then fused into $W_F(s)$
- How to compare $W_F(s)$ with $W_F(s')$? Could just use ℓ_2 (Matrix Frobenius Norm)
- Local delays (time warps) induce local perturbations in SSMs
- ℓ_2 norm unstable to these perturbations



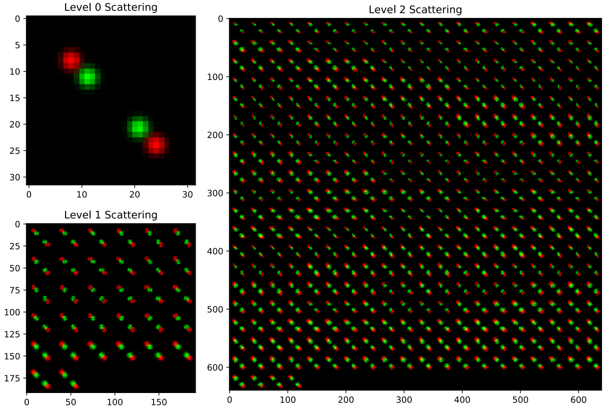
The Scattering Transform

- Instead of ℓ_2, use the *scattering transform* on SSMs
  - Has nice theoretical stability properties



The Scattering Transform: A Few Details

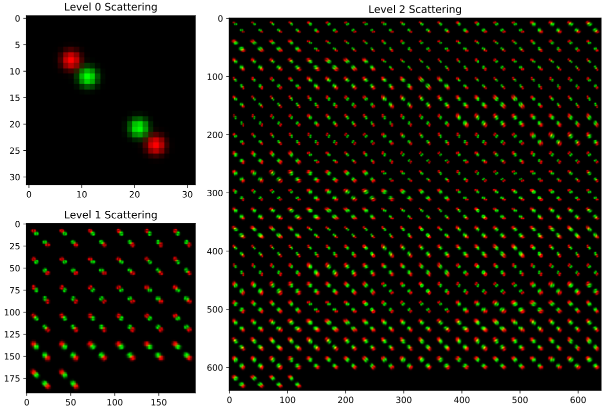
- Given an $N \times N$ image $I(u, v)$, choose lowpass filter $\phi(u, v)$
- Level 0: $S^0(u, v) = I * \phi(u, v)$
- There are $d \times d$ total coefficients: $d = N/2^{J-1}$, J max scale



The Scattering Transform: A Few Details

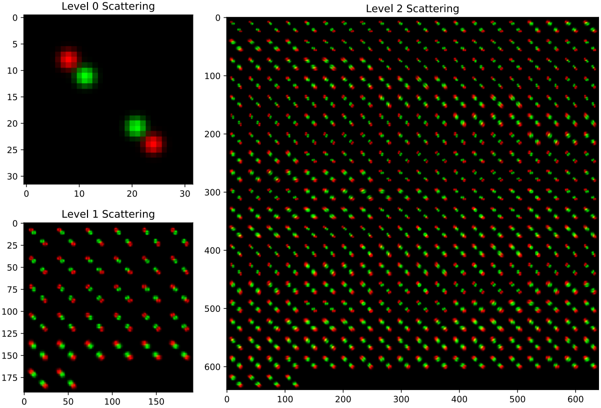
- Now choose a mother wavelet $\psi(u, v)$, a set of L directions γ_i, and a set of J scales $j \in 0, 1, \ldots, J-1$
- Level 1: $S^1_{i,j}(u, v) = |I * 2^{-2j}\psi_{\gamma_i}(u/2^j, v/2^j)| * \phi(u, v)$

  Using complex Gabor wavelets: $\psi_\gamma = e^{i\gamma\cdot(u,v)}e^{-(u^2+v^2)/\sigma^2}$



The Scattering Transform: A Few Details

- ▷ Now choose a mother wavelet $\psi(u, v)$, a set of L directions γ_i, and a set of J scales $j \in 0, 1, \ldots, J-1$
- ▷ Level 1: $S^1_{i,j}(u, v) = |I * 2^{-2j}\psi_{\gamma_i}(u/2^j, v/2^j)| * \phi(u, v)$
  There are d^2LJ level 1 coefficients

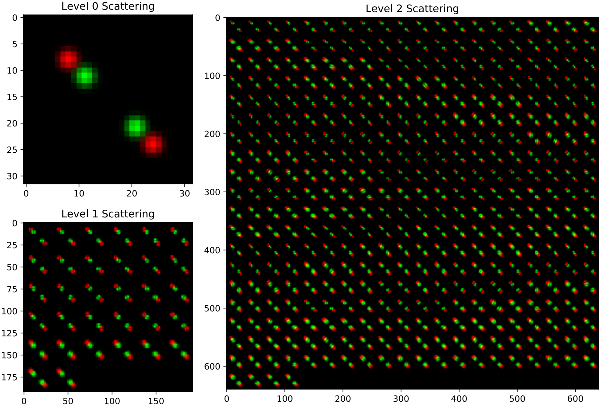


The Scattering Transform: A Few Details

- Level 2:

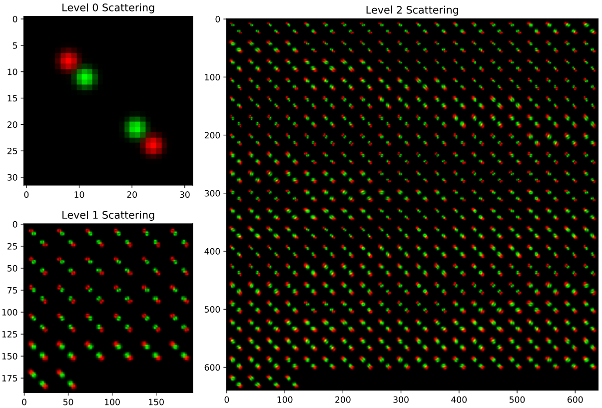
$$S^2_{i,j,k,l}(u,v) = ||I * 2^{-2j}\psi_{\gamma_i}(u/2^j, v/2^j)| * 2^{-2l}\psi_{\gamma_k}(u/2^l, v/2^l)| * \phi(u,v) \quad (1)$$

- There are $d^2 L^2 J(J-1)/2$ level 2 coefficients



The Scattering Transform: A Few Details

- One can continue past level 2, but we stop there
- Repeated convolve-with-wavelet, take complex modulus, do low-pass filter gives CNN-style architecture, but unsupervised.
- Each choice of wavelets in sequence is called a *path*

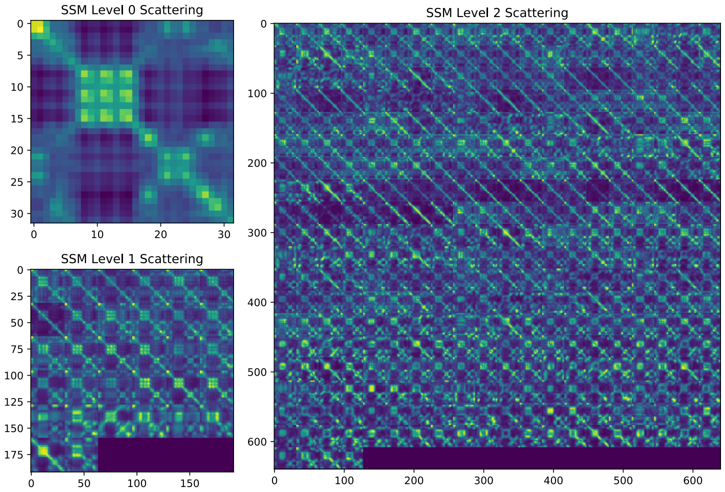


Scattering Transform As Feature Extractor

- Resize each SSM to 256×256 resolution
- Take $L = 8$ equally spaced directions between 0 and π
- Take $J = 4$ scales, so that each path is 32×32
- Results in $32^2(1 + 4 \times 8 + 8^2 \times 4 \times 3/2) = 427,008$ scattering coefficients extracted from SSM (6.5x data size, but stable)

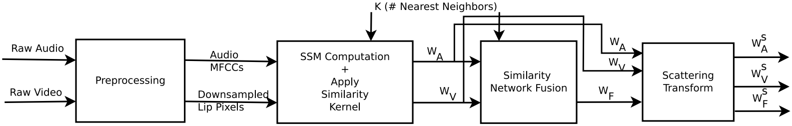
Scattering Transform As Feature Extractor

- Example scattering SSM

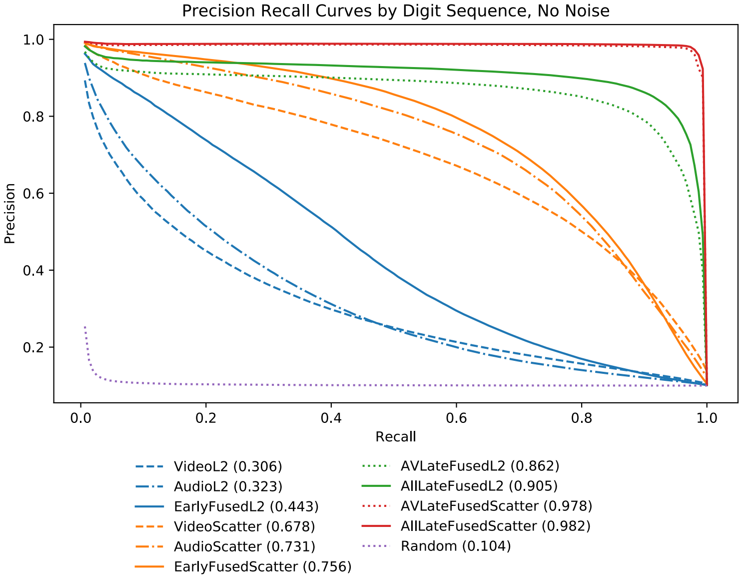


SNF for Late Audio-Visual Fusion

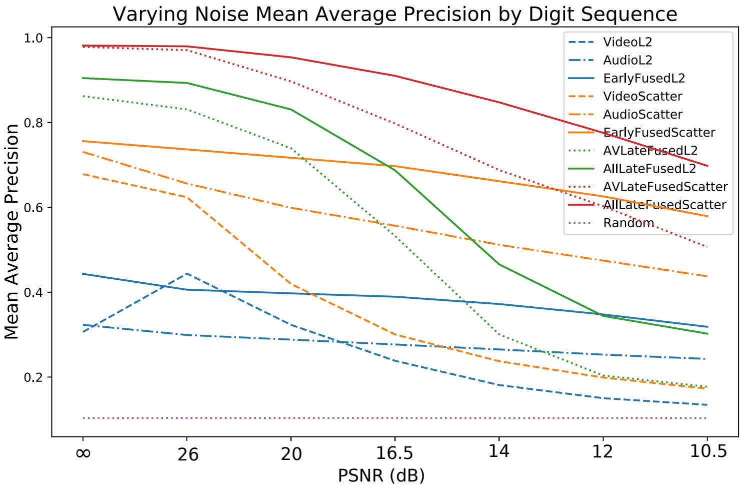
- Everything so far has happened *upstream*: before ranking decisions are made
- Can also apply SNF *downstream*
- Given object-level metrics $\mu_1, \ldots, \mu_k$ on set of N objects (strings)
- Each one produces *object-level* SSMs, which can themselves be fused into a new SSM
- We apply that here with $k = 3$ (audio, visual, early fused)



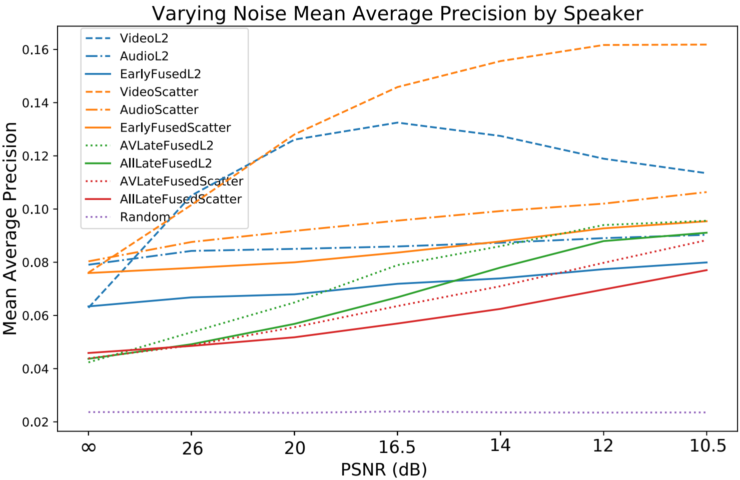
Results: Digit String Identification



Results: Digit String Identification, Simulated Noise



Results: Speaker Identification, Simulated Noise



Results: Joint Speaker And String Identification, Simulated Noise