Inference of scene representations for video analysis

Georgios Georgiadis

I

Video compression: a quick introduction

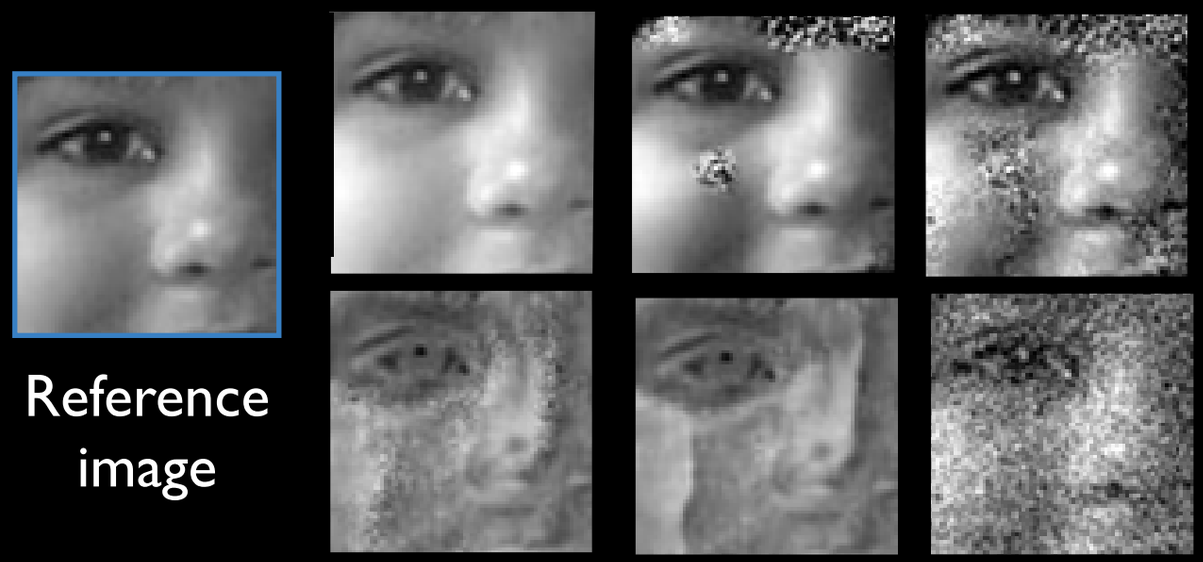
- GOAL: Reduce redundancy in video data
- Main techniques: Spatial image compression and motion compensation
- Lossless compression schemes is not necessary for video compression
- Lossy compression can produce perceptually indistinguishable images
- Trading off perceptual similarity with encoding bits

2

Perceptual similarity

- PSNR is ineffective in measuring perceptual similarity

All noisy images have the same PSNR value



Reference image

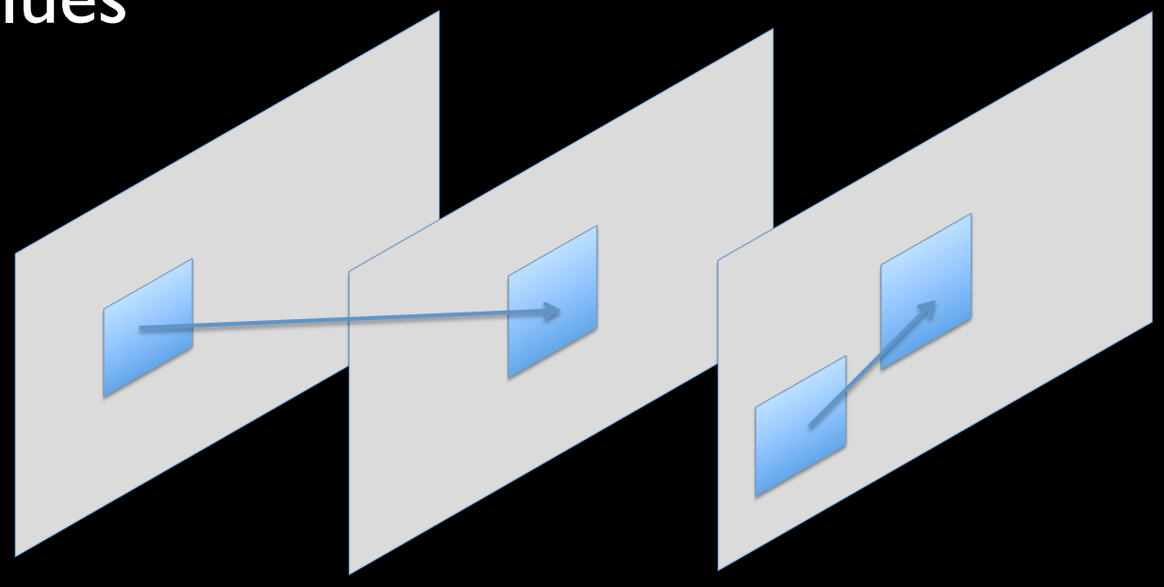
Wang et al., "Mean Squared Error: Love it or Leave it?", IEEE Signal Processing Magazine, 2009

- Assumption: We are given a metric that correctly measures the perceptual dissimilarity between two images or two videos
- What is the highest compression rate we can achieve for a given fidelity (in terms of perceptual similarity)

3

Modeling of pixel values

- Current encoders model pixel values
- Successful because:
    1. high compression ratio
    2. fast decoding
- Problem: Hit compression wall



Temporal prediction

Spatial prediction

- CISCO projects:
    1. By 2016: 1.2 million video minutes (>2 years) will travel the Internet every second
    2. Fastest consumer mobile service will be mobile video (from 271 million in 2011 to 1.6 billion in 2016)
    3. By 2016 video traffic will be 86% of global consumer traffic

4

Modeling the scene

- Key idea: Modeling the pixel values is not necessary to achieve perceptual indistinguishability

- The scene is an object that has a certain:
  - Photometry (appearance)
  - Dynamics (motion)
  - Geometry (shape)
  - Topology (occlusions)
  - Statistical regularities (e.g. textures)

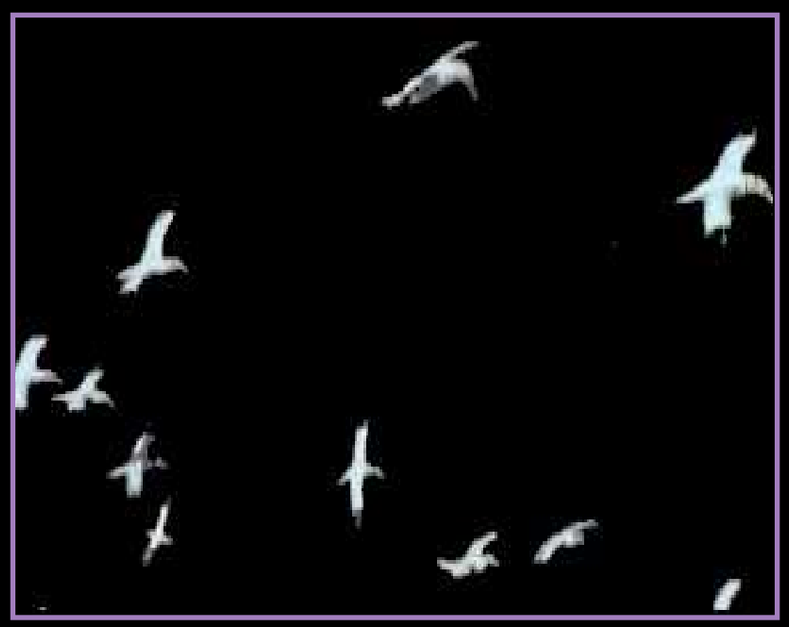
- Encoder should pay attention to visual phenomena the user is most sensitive (e.g. occlusions)

- Modeling the scene gives greater flexibility on how to spend encoding bits (compared to modeling the pixel values)
- Consequence: Ignore details that the viewer cannot perceive

5

Main approach
Texture / structure partition

Visual Structures



Temporal redundancy

Visual Textures

Spatial redundancy

6

Visual Textures

Regular textures

Stochastic textures

Domain-deformed textures

Range-deformed textures

- Definition: A texture is a region of an image that can be rectified into a sample of a stochastic process that is:
  - Locally stationary (defining characteristic)
  - Ergodic (allows inference from a single realization)
  - Markovian (allows finite dimensional model)

7

Visual textures
Stationarity and ergodicity

Notation:

$I \sim dP(I)$

$\phi : \mathbb{R}^N \to \mathbb{R}^W$ Statistic

$\phi_\omega \doteq \phi(I(\omega))$ "Local" Statistic

$\omega \subset \mathbb{Z}^2$

$I(\omega) \doteq \{I(x),\ x \in \omega\}$

Stationary process

$$\mathbb{E}(\phi_\omega) = \mathbb{E}(\phi_{\omega+T}), \quad T \in \mathbb{R}^2$$

Stationary and ergodic process

$$\frac{1}{N}\sum^{N} \phi_{\omega+T_i} \xrightarrow{\text{a. s.}} \mathbb{E}(\phi_\omega)$$

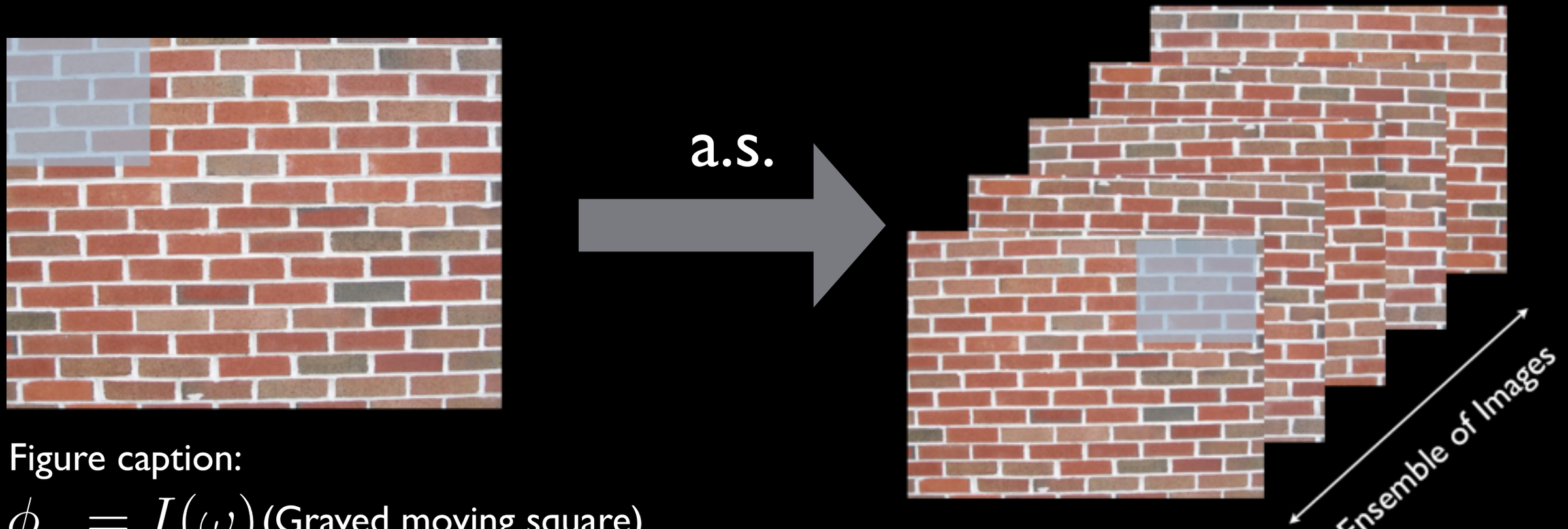


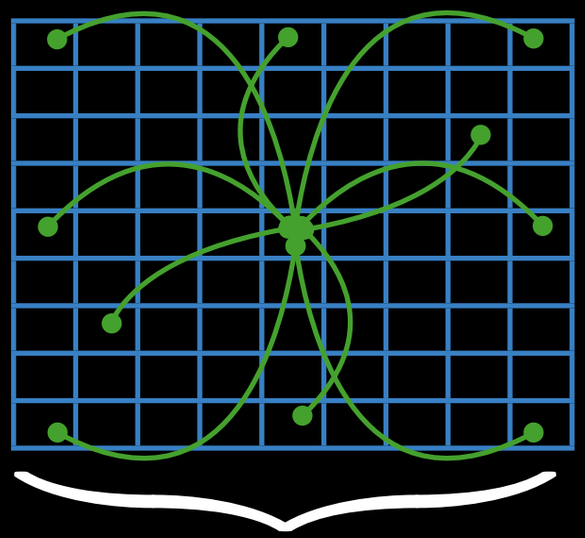
Figure caption:

$\phi_\omega = I(\omega)$ (Grayed moving square)

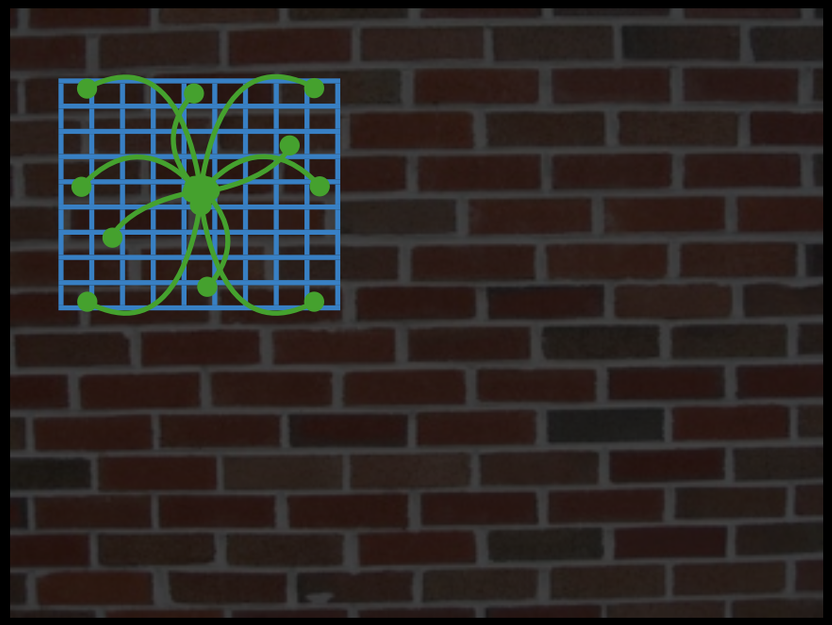
8

Visual Textures: Markovianity

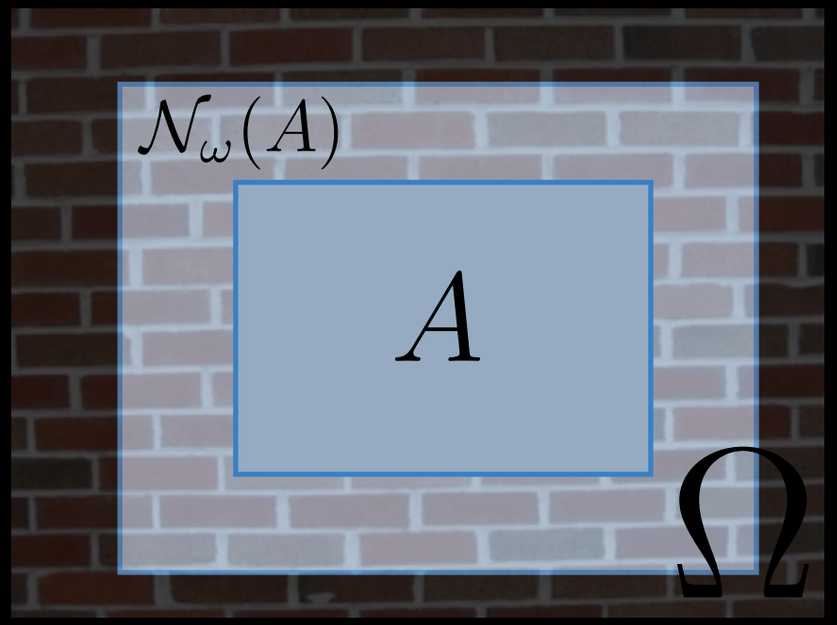
Neighborhood structure of x



$\omega_x \doteq \mathcal{N}(x)$



Homogeneity of neighborhood structure of x



A and its neighborhood N(A)

$$I(A) \perp \underbrace{I(\Omega - A)}_{\text{“outside”}} \mid \underbrace{I(\mathcal{N}_\omega(A))}_{\text{“inside”}}, \forall A \subset \Omega$$

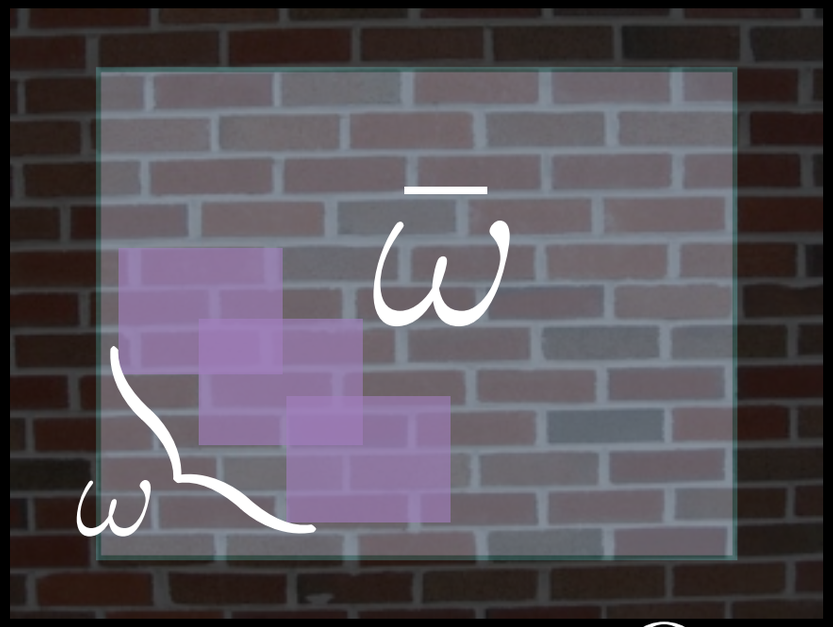
$$H\left[I(A)|I(\mathcal{N}_\omega(A))\right] = H\left[I(A)|I(\Omega - A)\right]$$

$$\hat{\omega}(\beta) = \arg\min_{\omega} H\left[I(A)|I(\mathcal{N}_\omega(A))\right] + \frac{1}{\beta}|\omega|$$

Visual Textures Representation

$$H\left[I(A)|I(\mathcal{N}_\omega(A))\right] \simeq -\frac{1}{M}\sum_{i=1}^{M}\log p\left[I(A_i)|I(\mathcal{N}_\omega(A_i))\right]$$

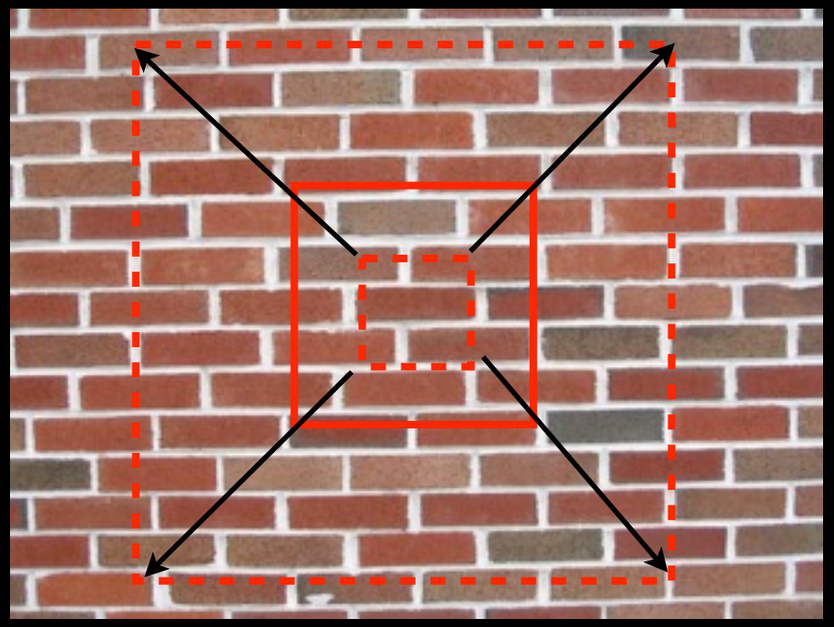
$$A_i \doteq A + T_i \subset \Omega$$

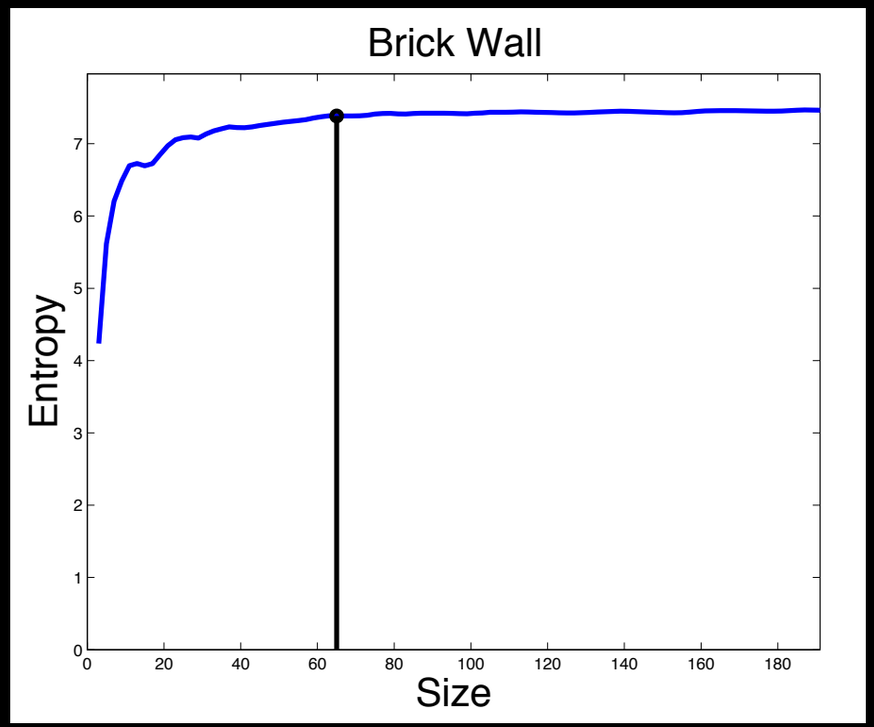


$$\omega \subset \bar{\omega} \subset \Omega$$

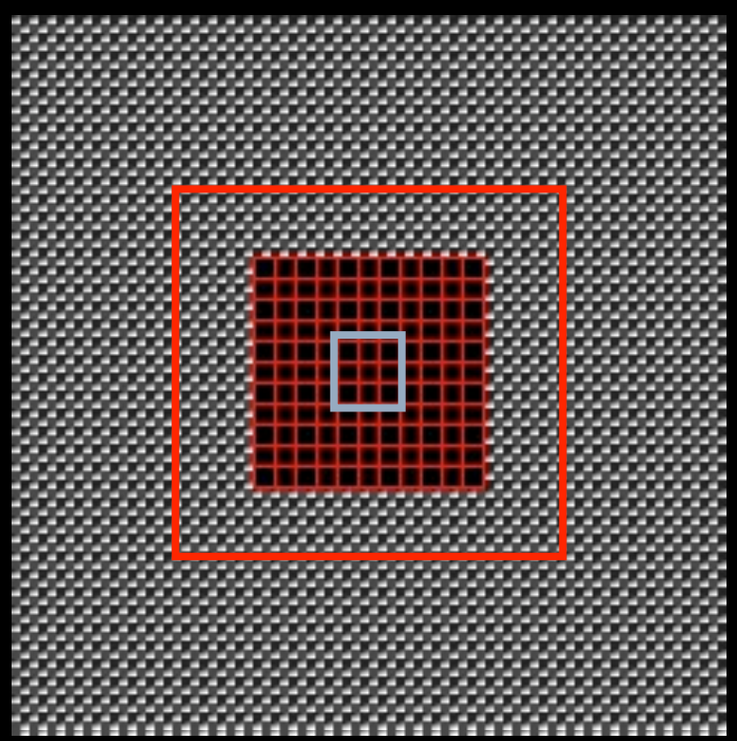
$$\hat{r}, \hat{\sigma} = \arg\min_{r=|\omega|,\sigma=|\bar{\omega}|} \hat{H}\left[I(A)|I(\mathcal{N}_\omega(A))\right] + \frac{1}{\beta}|\bar{\omega}|$$

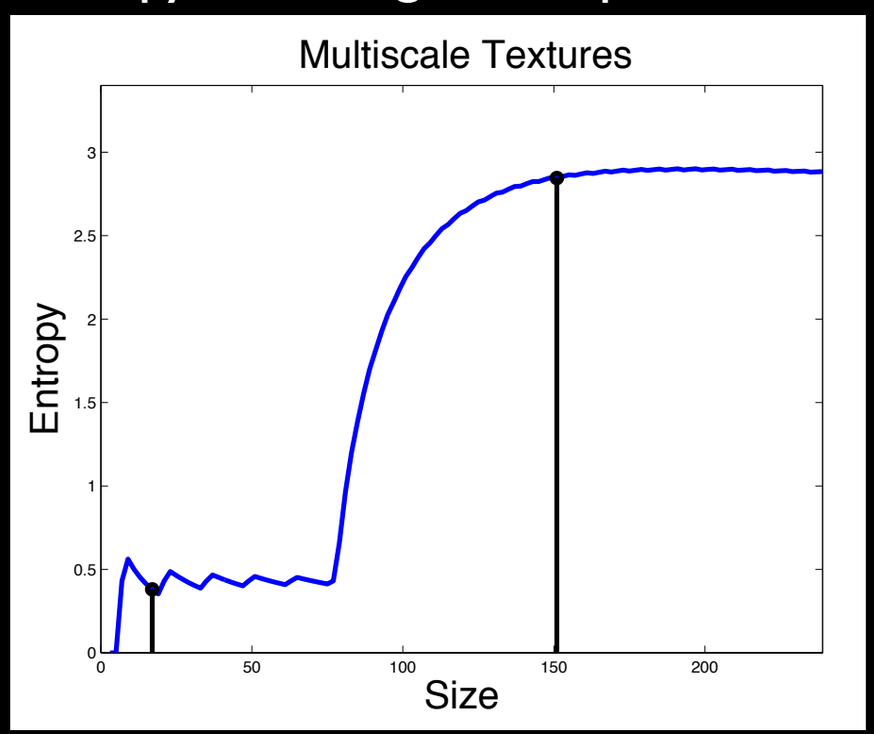
Visual Textures
Inferring ω





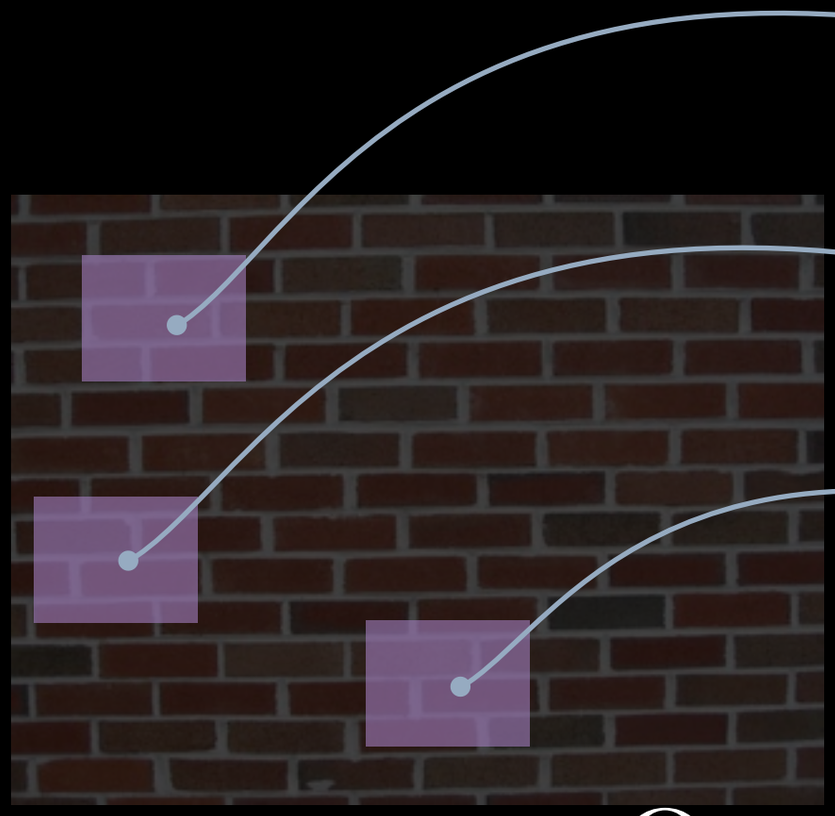
Entropy of histogram of pixel values



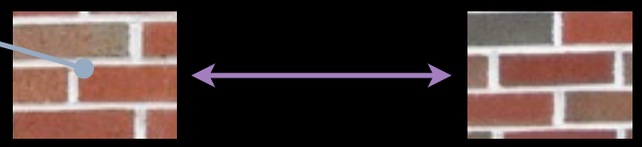
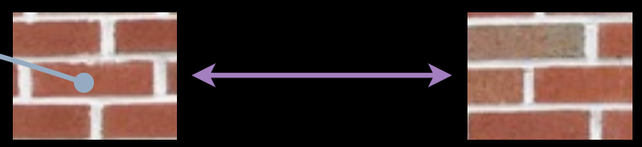
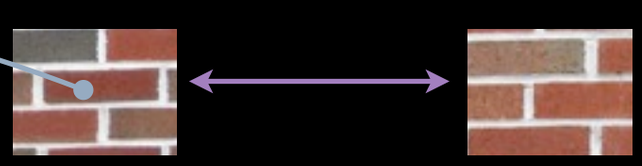


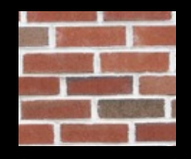
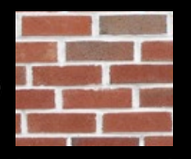
11

Visual Textures
Inferring $\bar{\omega}$



$$\omega \subset \bar{\omega} \subset \Omega$$

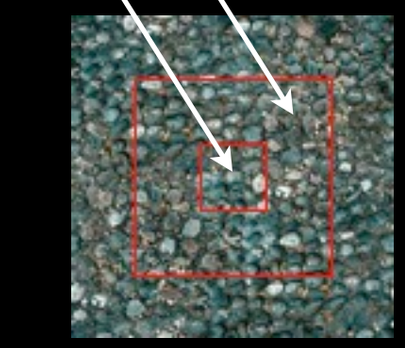
Sampled Patches	Nearest Neighbor of sampled patch	Distance	Decision
		$< \varepsilon$	Keep
		$> \varepsilon$	Discard
		$< \varepsilon$	Keep

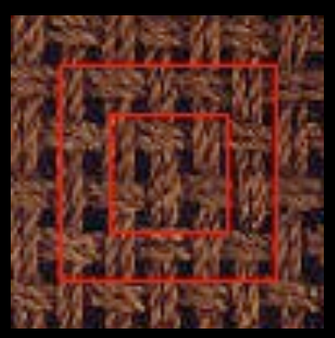
$$\bar{\omega} = \{ \underbrace{\text{  }}_{\omega}, \underbrace{\text{  }}_{\omega} \}$$

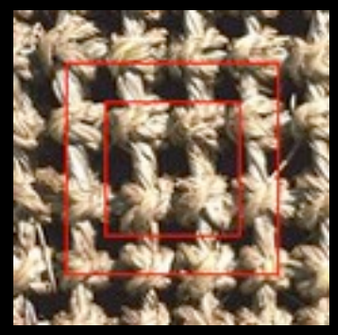
Resample a squared region of same size as $|\bar{\omega}|$

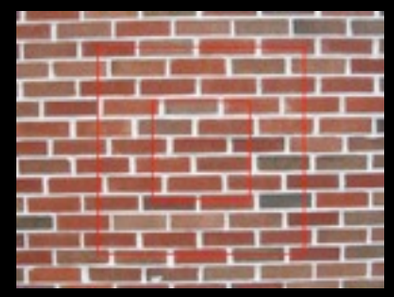
12

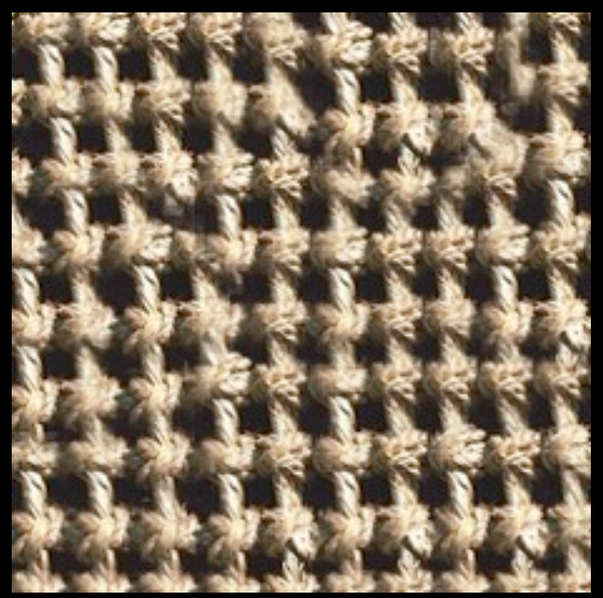
Visual Textures
Inferring representations - Results

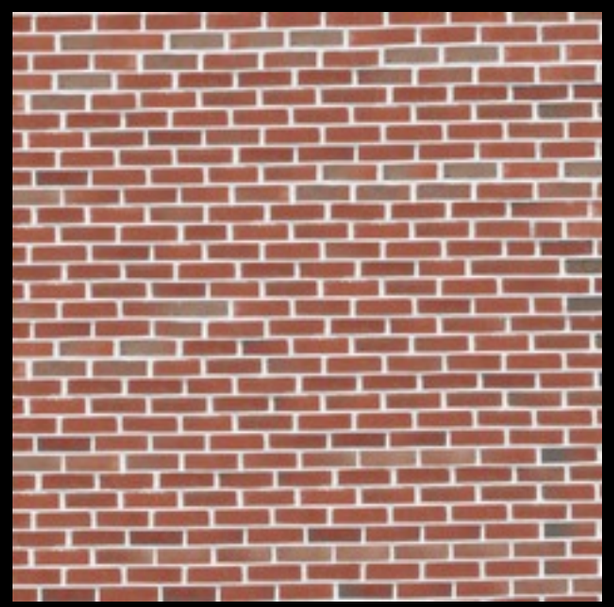












Synthesized textures from the representation

$$E(\hat{I}_s; I(\bar{\omega})) = \sum_{\tilde{\omega} \in \hat{\Omega}} \| \hat{I}_{\tilde{\omega}} - I_{\tilde{\omega}} \|^2$$

Synthesized texture

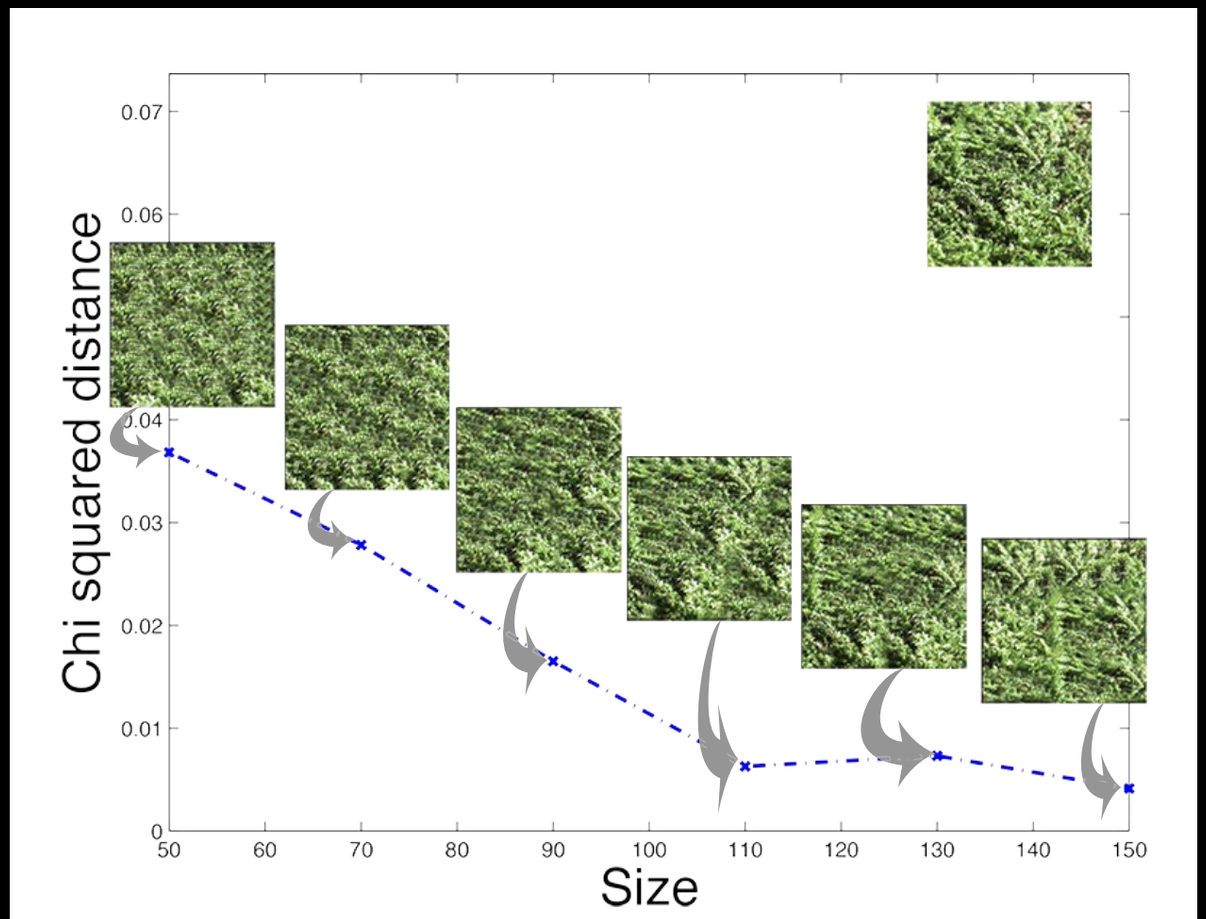
Texture representation

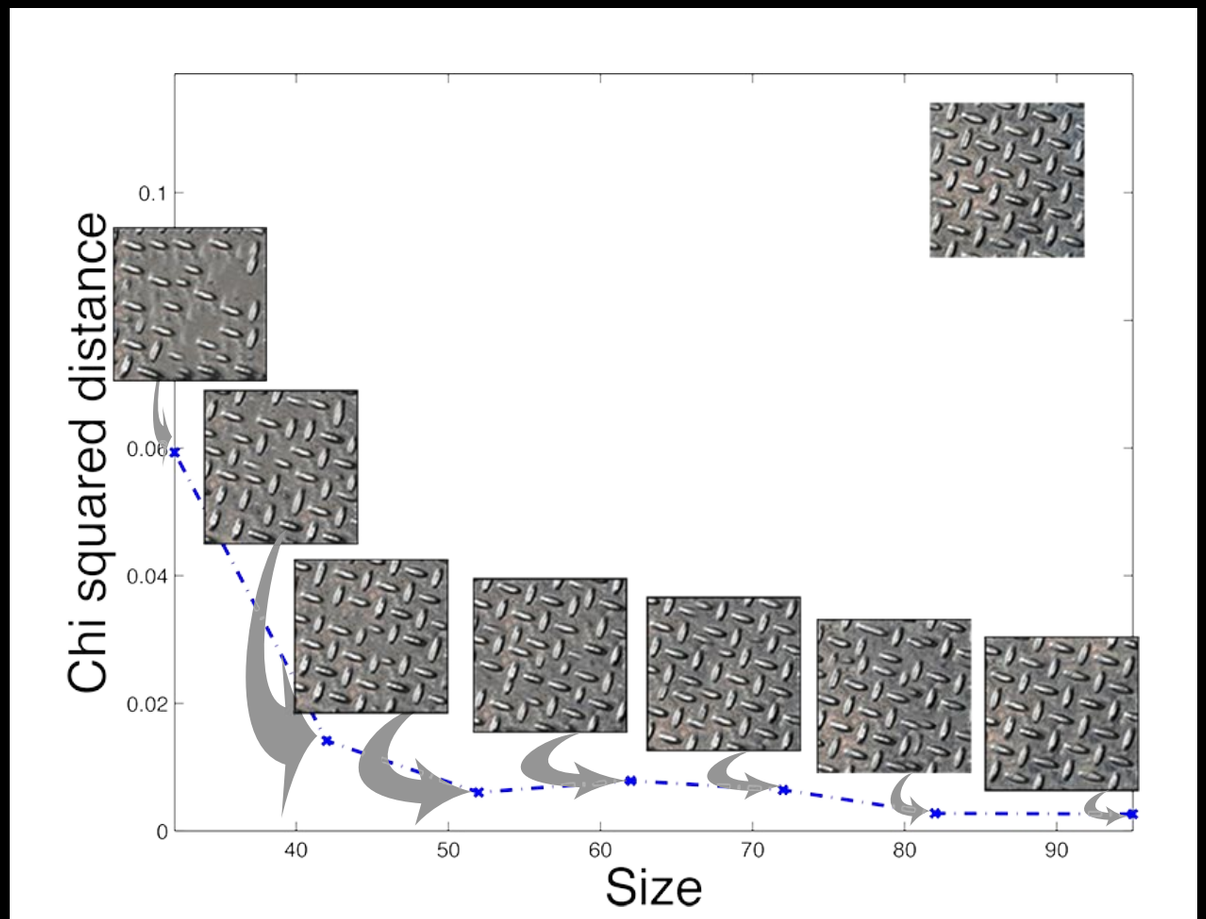
Local neighborhood $\tilde{\omega}$ in synthesized texture

Nearest neighbor of $\tilde{\omega}$ in the representation

13

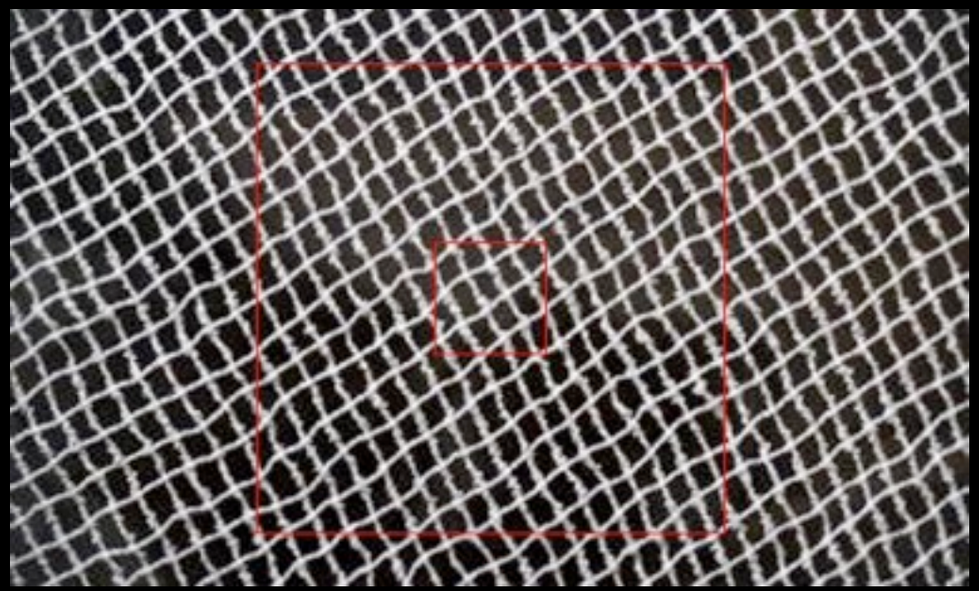
Visual Textures
Rate - Distortion plots

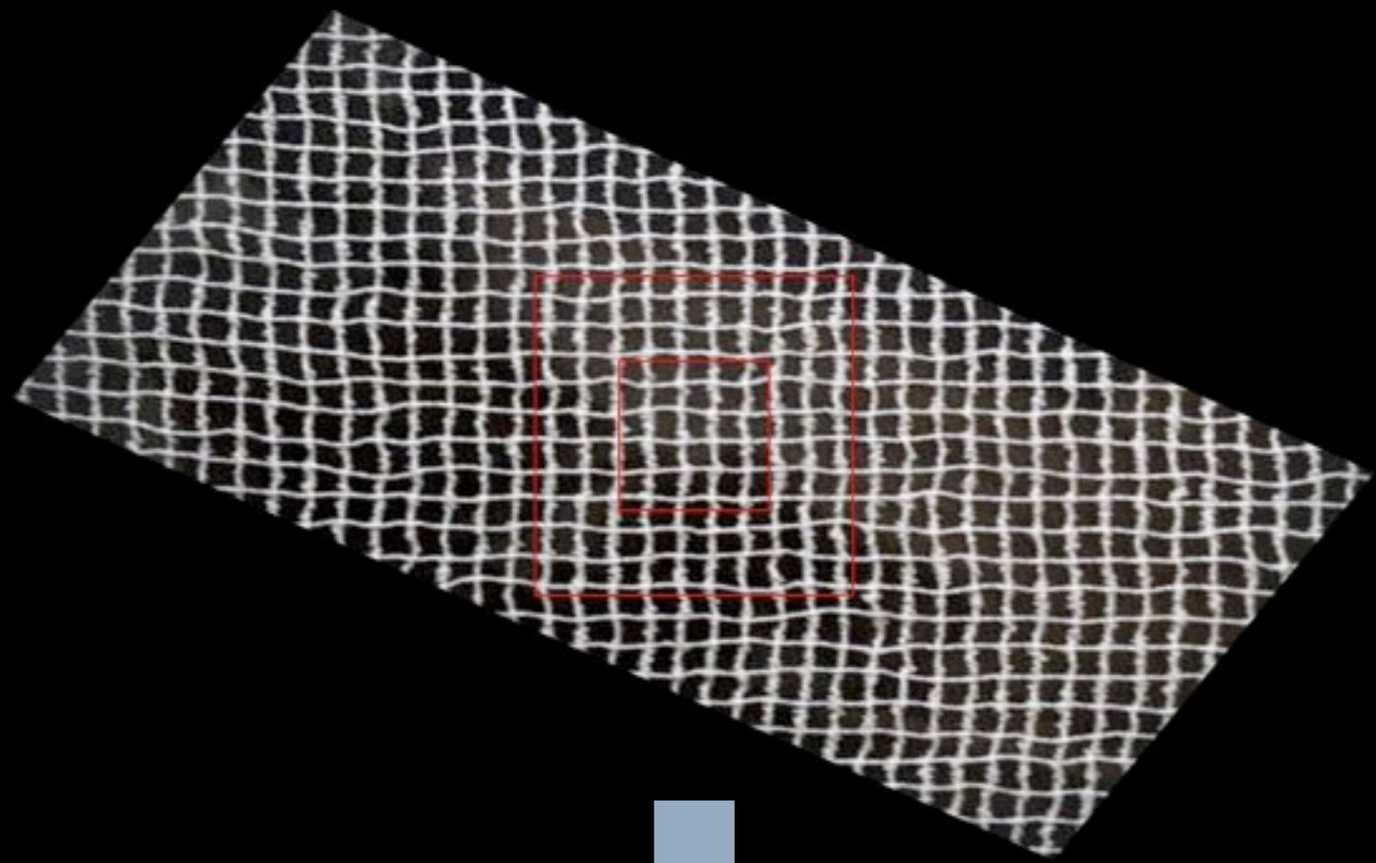




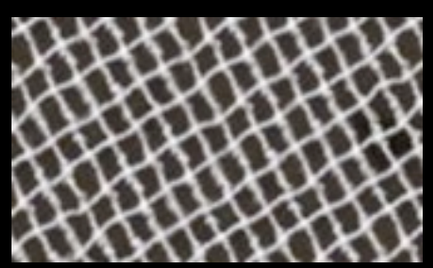
14

Visual Textures
Transformations

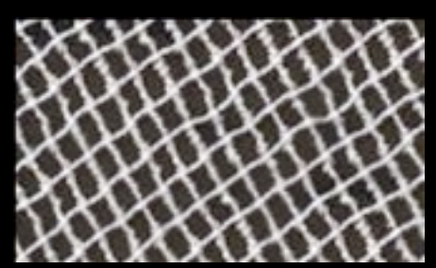




$$|\bar{\omega}_{rectified}| \simeq |\frac{\bar{\omega}_{original}}{4}|)$$



Synthesized texture



Synthesized texture

15

Visual Textures Segmentation

Algorithm:

- Image is divided into small regions
- Sequentially merge regions that are the most "similar" (their pixel value histograms are the most similar)

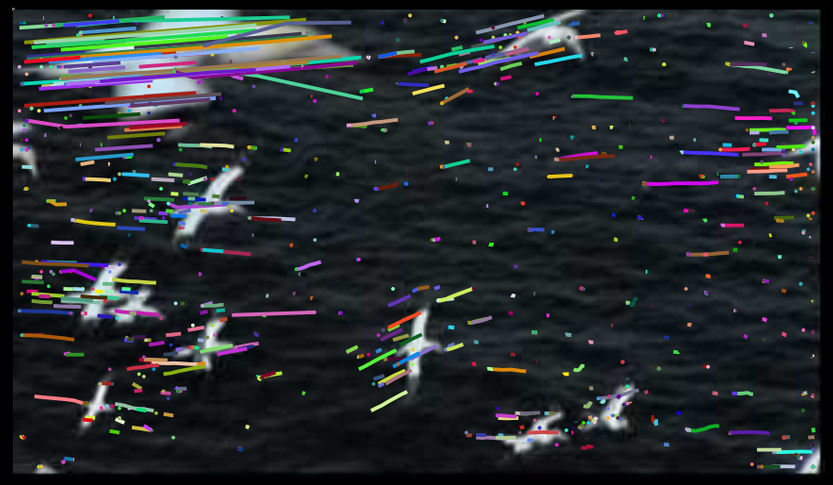
Visual Structures

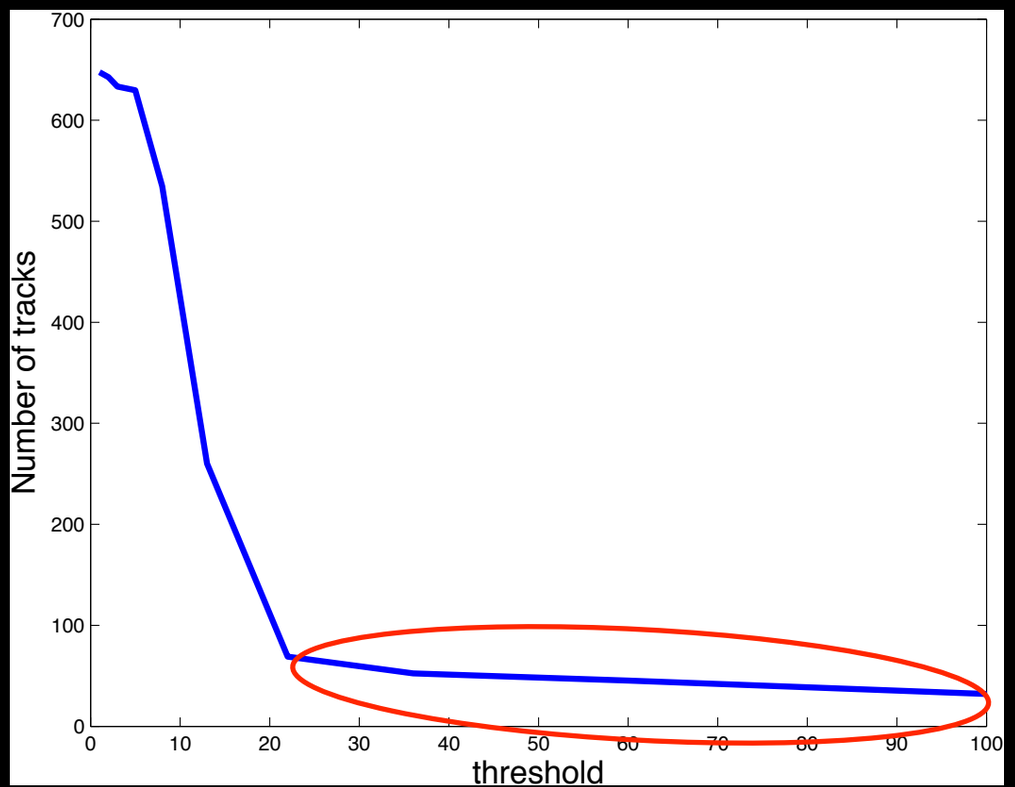
Visual Structures

Temporal redundancy

- Structures are regions in images that:
  - trigger isolated responses of a feature detector
  - persist in more than one image
- Structures ⟷ Saliency

Visual Structures
Detection, tracking and selection



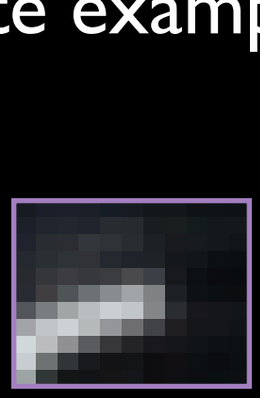


- Most reliable tracks persist over a wide range of threshold values:
  - stable reference frame (high quality)
  - large temporal support (highly compressible)

18

Visual Structures Representation

Template examples:



- Dictionary is composed of templates that have different spatial resolutions and are functions of different temporal supports
- Final representation: template and pixel location of central pixel

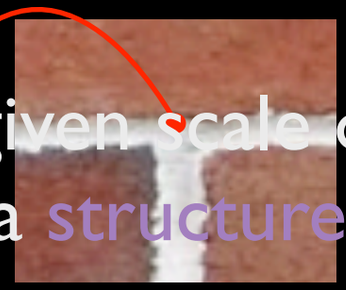
19

Texture / Structure partition

- Texture / Structure partition depends on multiple frames

- Texture / Structure partition depends on scale

Finer Scale: Structure



Coarser Scale: Texture

- For any given scale of observation σ, a region $\bar{\omega}$ with $|\bar{\omega}| = \sigma$ is either a structure or a texture.

Isolated Extremum at scale ε
=> Not Stationary
=> Not Texture

Stationary at scale σ >> ε
=> No isolated extremum at scale σ
=> Not a structure

20

Texture / Structure partition
A first attempt

Input frame

Structures

Texture segmentation

Texture compression

Texture synthesis

Reconstructed frame

Modeling the scene

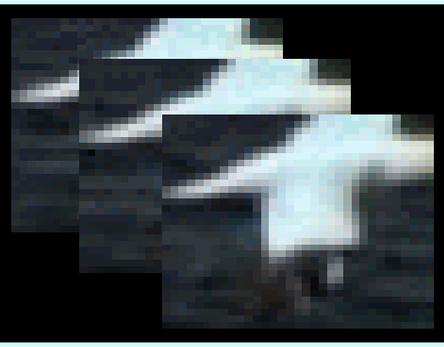
Determine structures

Texture segmentation

Texture compression

Texture synthesis

Calculate template dictionary

Region type	Redundancy	Calculate	Representation
Structure			Template, position
Texture			Compressed representation, texture boundary

22

Texture / Structure partition
Qualitative results

Original video

Reconstructed video

23

Comparison with other methods

- Our method successfully captures salient regions of the video

Original

Zhi et al. [5] - ICCV'11

Our method

24

Computational cost

- For a 332x192 resolution video (7 frames):
  - Tracking (once for whole video): ~2s
  - Texture segmentation (per frame): ~1m
  - Texture compression (once per texture): ~45s
  - Texture synthesis (once per texture): ~3m

- Overall:
  - Encoding time per frame: ~66s
  - Decoding time per frame: ~30s (texture synthesis)

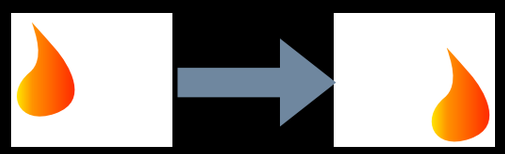
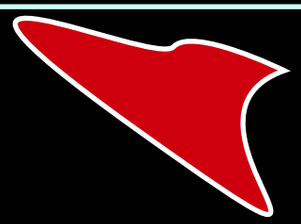
25

Modeling the scene (a year before)

Calculate optical flow and occlusions → Calculate tracks

Segmentation

→ Texture / Structure encoding

Region type	Redundancy	Calculate	Encode
Structure		Velocities	Median motion vector
Texture		Crop / Downsample	
Occlusions/ Homogeneous	? 	Intensities	Mean

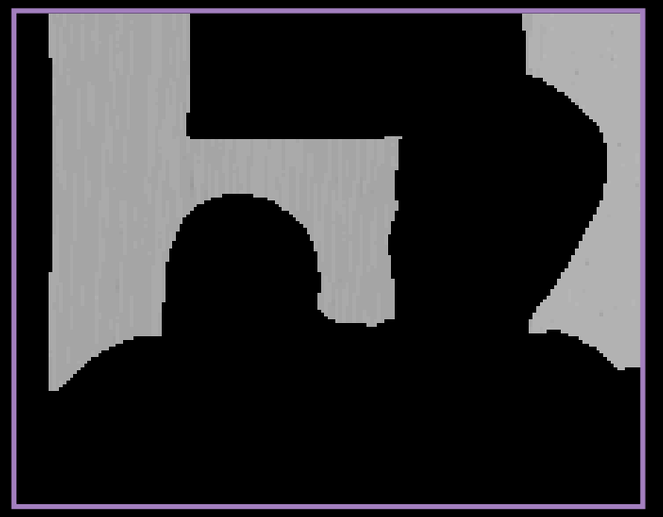
26

Temporal / Spatial / Textural partition

Input frame

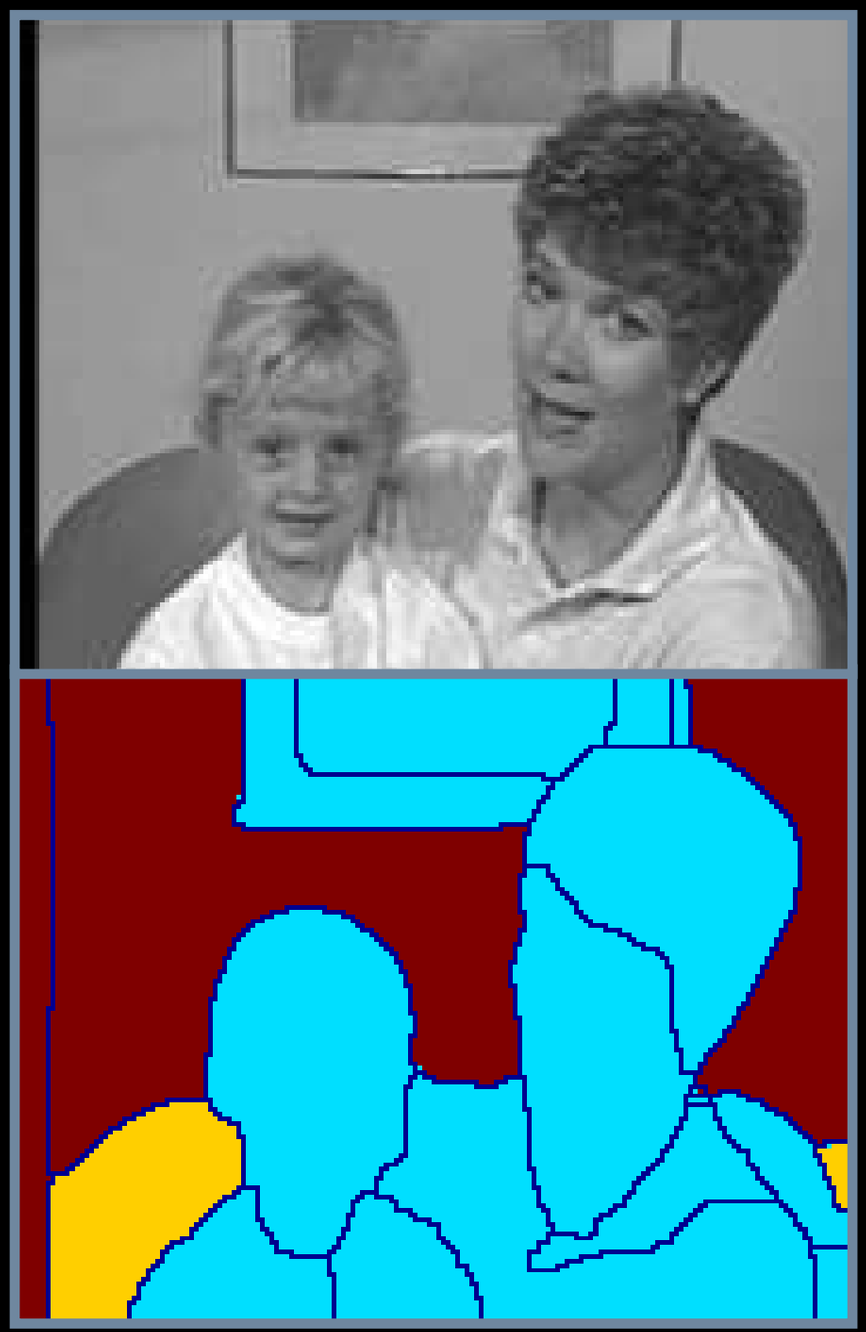
Temporal prediction

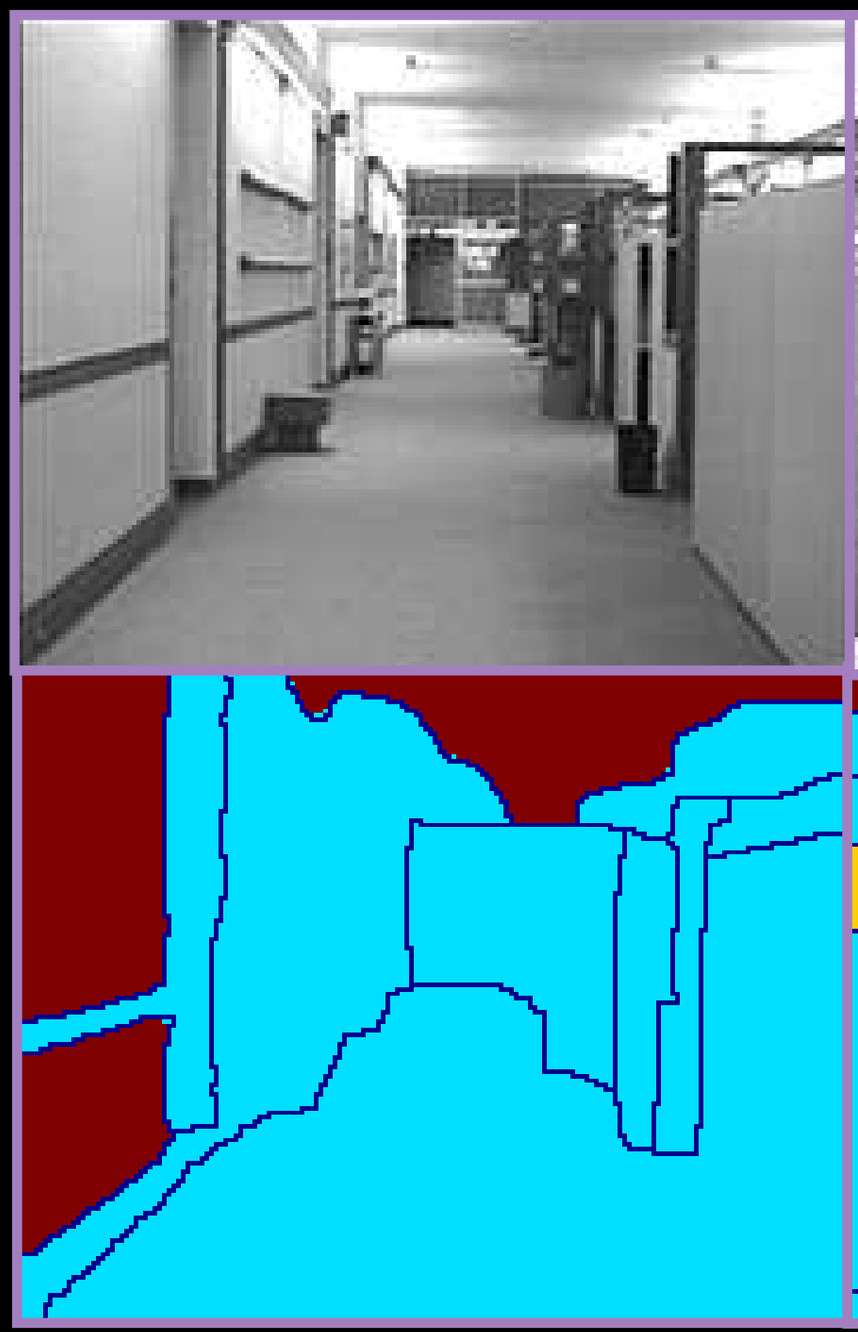
Spatial prediction

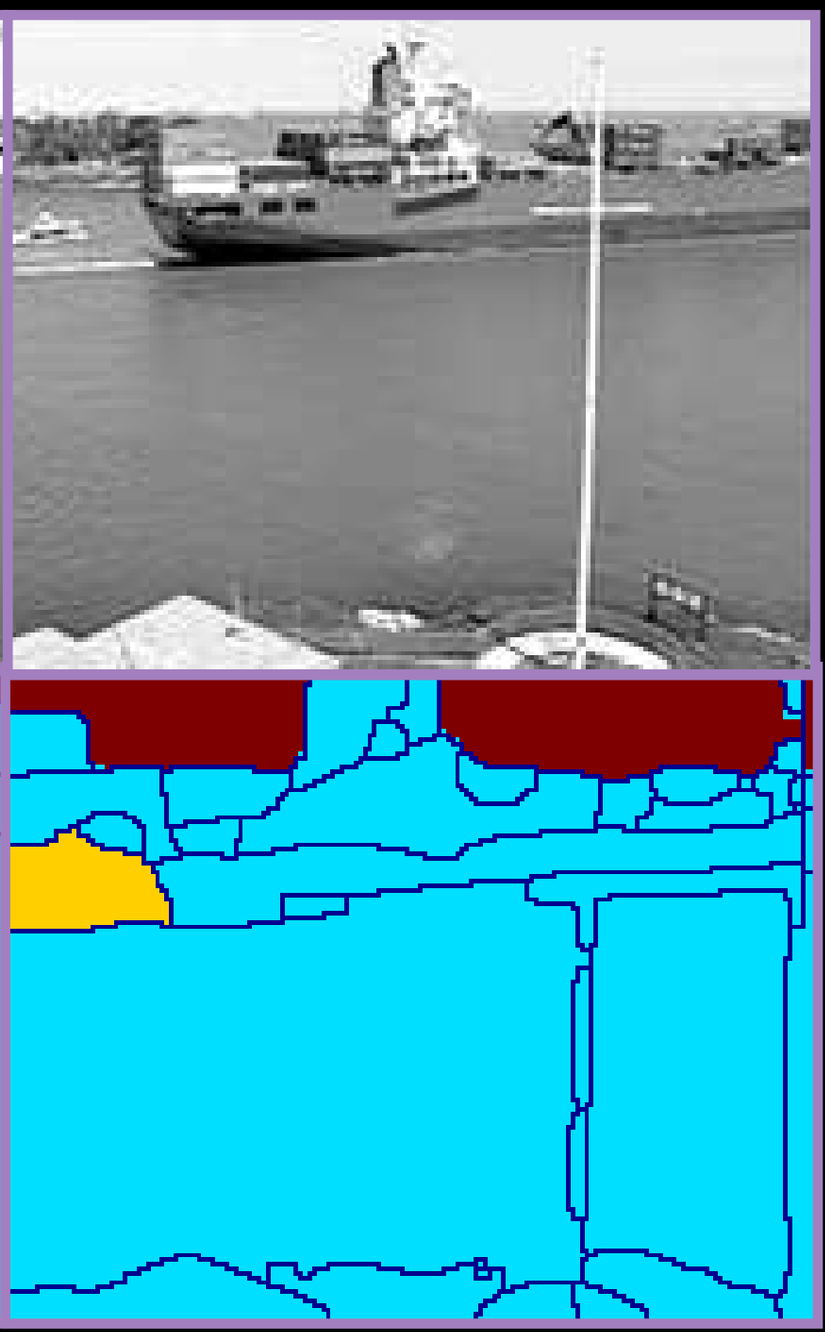


Texture synthesis

Partition results







Blue: Temporal prediction
Yellow: Spatial prediction
Red: Texture synthesis

Failure cases

28

Qualitative results

Predicted frame Error frame Reconstructed frame

29

Original video

Our method

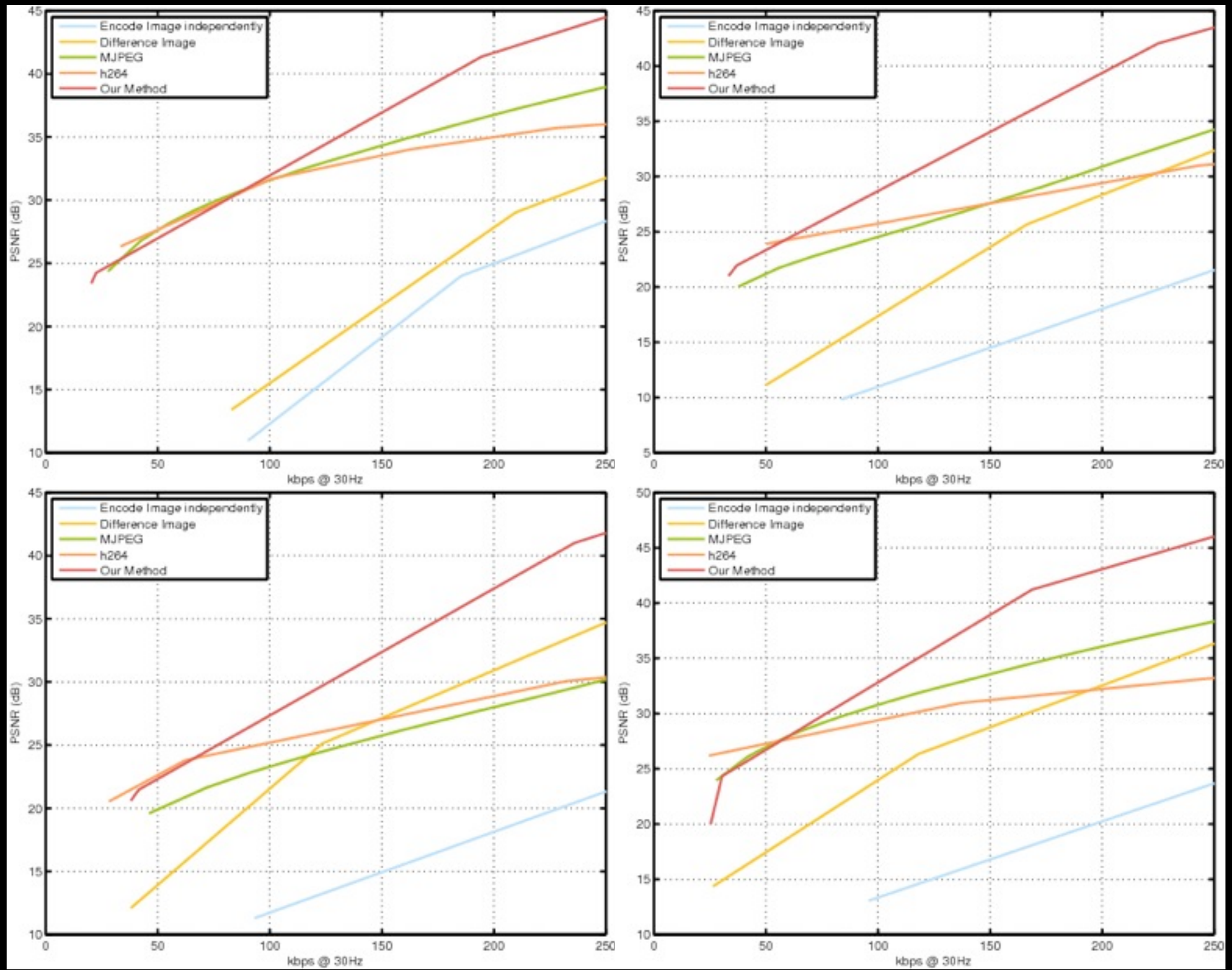
Same file size

H.264

Reconstructed videos

30

Quantitative results



31

Computational cost

- For a VGA resolution video:
  - GPU implementation of optical flow: ~10s
  - Tracking: ~2s
  - GPU implementation of segmentation: ~13s
  - Texture synthesis: ~5s
- Segmentation can run in parallel with optical flow and tracking
- Encoding time per frame: ~21s

32

Actionable Saliency Detection

Occlusion detection:

- Detect salient regions as violations of co-visibility hypothesis:

$$H0 = \{\exists \text{ a diffeo } w : \Omega \to D \mid I(x, t + dt) - I(w(x), t) \overset{\text{IID}}{\sim} \mathcal{N}\}$$

- No violation when moving over homogeneous backgrounds

33

Approach

Co-visibility hypothesis:

$$H0 = \{\exists \text{ a diffeo } w : \Omega \to D \mid I(x, t + dt) - I(w(x), t) \overset{\text{IID}}{\sim} \mathcal{N}\}$$

Rigidity:

$$H1 = \{\exists\, T \in \mathbb{R}^3, \omega \in \mathbb{R}^3,\ Z : D \to \mathbb{R}^+ \mid w(x) = \pi(\exp(\widehat{\omega})\bar{x}Z(x) + T)\}$$

Notation:

$x \in \mathbb{R}^2$	Pixel location
$y(x) \in \mathbb{R}^2$	Motion vector
$V \in \mathbb{R}^3$	Translational velocity of camera
$\omega \in \mathbb{R}^3$	Rotational velocity of camera
$Z : D \to \mathbb{R}^+$	Depth map

$$y(x) = \mathcal{A}(x)\frac{V}{Z(x)} + \mathcal{B}(x)\omega$$

$$\underset{V}{\text{minimize}} \quad \psi(V) = \frac{1}{2}||\hat{C}(V)Y||_2^2.$$

$$\left[I - \mathcal{W}C(C^T\mathcal{W}^T\mathcal{W}C)^{-1}C^T\mathcal{W}^T\right]\mathcal{W}Y \doteq \hat{C}(V)Y = 0.$$

34


STOP
STOP

Results

36

Future work

- Structures: template selection

- Textures:

  (i) Texture segmentation (>2 regions)

  (ii) Texture synthesis: speed-up?

  (iii) Dynamic textures

  (iv) Structured textures

- Texture / Structure partition

37

Conclusions

- Model the scene rather than the pixel values

- We provided definitions, developed representations and designed algorithms for textures and structures

- Next generation video compression schemes need to model the scene, semantics, context. Encoding should depend on the application, the viewer and many more

38

Questions?