Compressed Sensing and Dictionary Learning to Alleviate Tradeoff between Temporal and Spatial Resolution in Videos

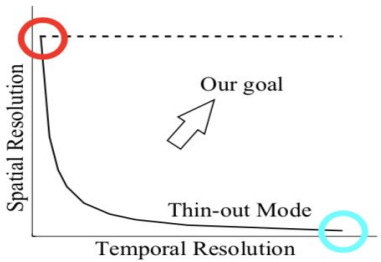
EE 771 Course Project

Karan Taneja (15D070022)
Anmol Kagrecha (15D070024)
Pranav Kulkarni (15D070017)

Contents

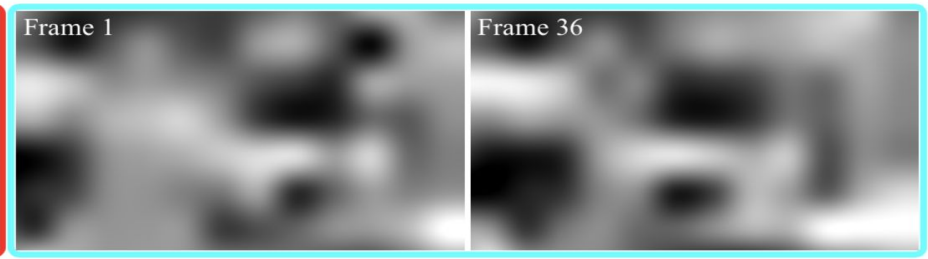
Problem Statement

Fundamental trade-off in cameras is due to hardware factors such as readout and analog-to- digital (AD) conversion time of sensors



(a) Resolution trade-off

(b) Motion blurred image



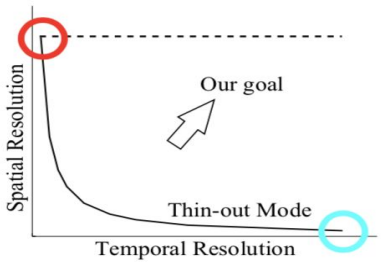
(c) Thin-out mode: Low spatial resolution, high frame rate

Problem Statement

Fundamental trade-off in cameras is due to hardware factors such as readout and analog-to- digital (AD) conversion time of sensors

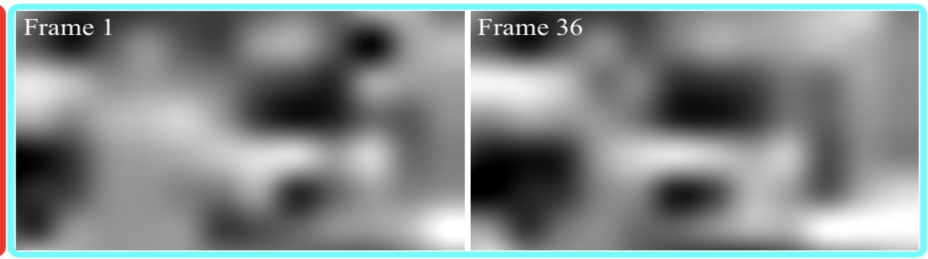
Solution: Parallel AD convertors and frame buffers - incurs more cost!

'Thin-out' mode (high speed draft): directly trades off the spatial resolution for higher temporal resolution and often degrades image quality



(a) Resolution trade-off


(b) Motion blurred image



(c) Thin-out mode: Low spatial resolution, high frame rate

Overcome this tradeoff without incurring a significant increase in hardware costs.

Overview of the approach

- Exploit the sparsity of natural videos through framework of compressed sensing

Overview of the approach

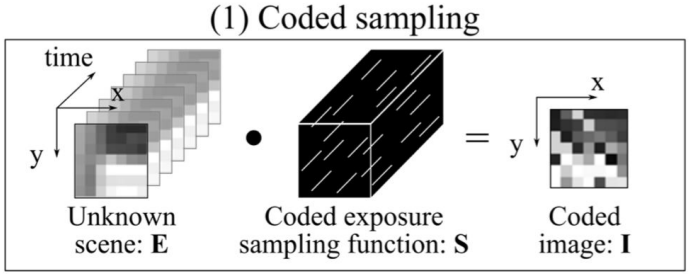
- Exploit the sparsity of natural videos through framework of compressed sensing
- Sampling: Sample space-time volumes while accounting for the restrictions imposed by imaging hardware
- Dictionary Learning: learning an over-complete dictionary from a large collection of videos, and represent any given video as a sparse, linear combination of the elements from the dictionary

Overview of the approach

- Exploit the sparsity of natural videos through framework of compressed sensing
- Sampling: Sample space-time volumes while accounting for the restrictions imposed by imaging hardware
- Dictionary Learning: learning an over-complete dictionary from a large collection of videos, and represent any given video as a sparse, linear combination of the elements from the dictionary

- Dictionary captures moving edges
- Overcomplete dictionary leads to sparse representation of videos
- Reconstruction: Solve inverse problem to get coefficients of the video in the learnt dictionary basis

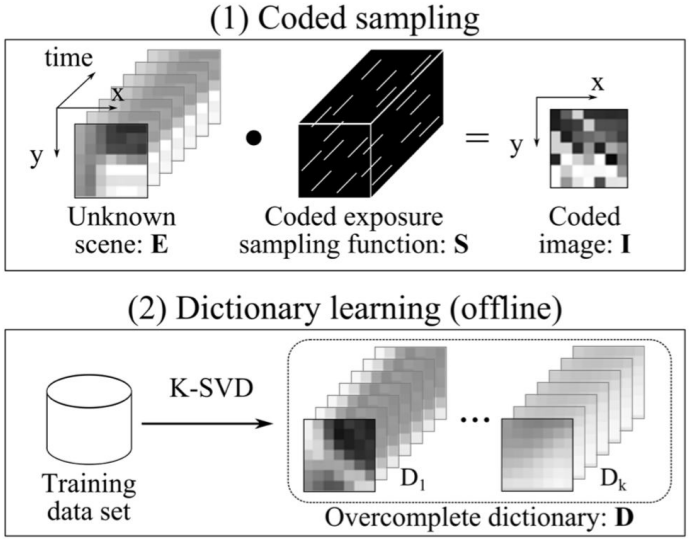
Overview of the approach



CMOS sensors with per pixel exposure, current architecture allows only a single bump (on-time) during one camera exposure.

Reconstruct all sub-frames from the coded snapshot

Overview of the approach

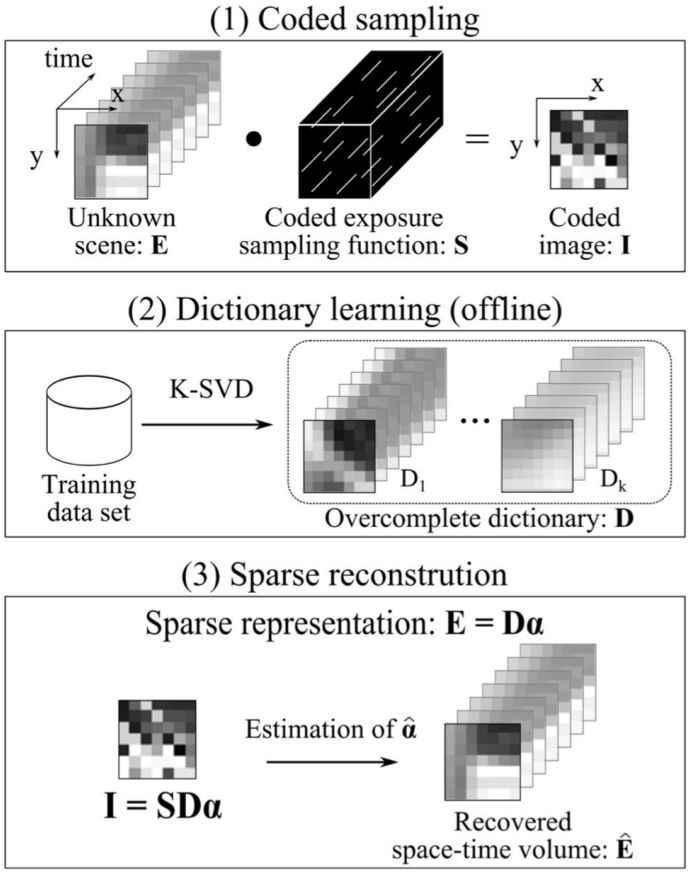


CMOS sensors with per pixel exposure, current architecture allows only a single bump (on-time) during one camera exposure.

Reconstruct all sub-frames from the coded snapshot

K-SVD used to learn a over-complete dictionary basis which allows sparse representation of videos in the dictionary basis.

Overview of the approach



CMOS sensors with per pixel exposure, current architecture allows only a single bump (on-time) during one camera exposure.

Reconstruct all sub-frames from the coded snapshot

K-SVD used to learn a over-complete dictionary basis which allows sparse representation of videos in the dictionary basis.

Recover the space-time volume from a single captured image. Use the learned dictionary and sampling matrix to get all subframes by using OMP for sparse signal recovery.

Coded Sampling

Hardware restrictions

- *Binary shutter:* Each pixel either collecting light or not at every instant
- *Single bump exposure:* only one continuous ‘on’ time
- *Fixed bump length:* for all pixels, limited dynamic range of sensors

Coded Sampling

Hardware restrictions

- *Binary shutter:* Each pixel either collecting light or not at every instant
- *Single bump exposure:* only one continuous 'on' time
- *Fixed bump length:* for all pixels, limited dynamic range of sensors

Coded image is $I(x,y) = \sum_{t=1}^{N} S(x,y,t) \cdot E(x,y,t)$

Where *E(x, y, t)* is space time volume, *S(x, y, t)* is per pixel shutter function

For conventional capture, *S(x, y, t) = 1 for all x, y, t.*

Dictionary Learning

$$\mathbf{E} = \mathbf{D}\boldsymbol{\alpha} = \alpha_1 \mathbf{D}_1 + \cdots + \alpha_k \mathbf{D}_k.$$

where

$\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_k]^T$ is the sparse vector coefficient

$\mathbf{D}_1, \cdots, \mathbf{D}_k$ are the dictionary elements

Dictionary Learning

$$\mathbf{E} = \mathbf{D}\boldsymbol{\alpha} = \alpha_1 \mathbf{D}_1 + \cdots + \alpha_k \mathbf{D}_k.$$

where

$\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_k]^T$ is the sparse vector coefficient

$\mathbf{D}_1, \cdots, \mathbf{D}_k$ are the dictionary elements

Algorithm used: K-SVD

No. of training videos: 20, rotated in 8 directions

Dictionary Learning

$$\mathbf{E} = \mathbf{D}\boldsymbol{\alpha} = \alpha_1 \mathbf{D}_1 + \cdots + \alpha_k \mathbf{D}_k.$$

where

$\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_k]^T$ is the sparse vector coefficient

$\mathbf{D}_1, \cdots, \mathbf{D}_k$ are the dictionary elements

Algorithm used: K-SVD

No. of training videos: 20, rotated in 8 directions

Finally, dictionary elements from all images are appended.

Sparse Reconstruction

Combining sampling and coded mage equation s in a vector form we have

$$\mathbf{I} = \mathbf{S}\,\mathbf{D}\,\alpha$$

Sparse Reconstruction

Combining sampling and coded mage equation s in a vector form we have

$$\mathbf{I} = \mathbf{S}\,\mathbf{D}\,\boldsymbol{\alpha}$$

Estimate of the coefficient vector is given by

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0 \quad \text{subject to} \quad \|\mathbf{SD}\boldsymbol{\alpha} - \mathbf{I}\|_2^2 < \epsilon$$

Sparse Reconstruction

Combining sampling and coded mage equation s in a vector form we have

$$\mathbf{I} = \mathbf{S}\,\mathbf{D}\,\boldsymbol{\alpha}$$

Estimate of the coefficient vector is given by

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0 \quad \text{subject to} \quad \|\mathbf{SD}\boldsymbol{\alpha} - \mathbf{I}\|_2^2 < \epsilon$$

OMP is used to find these estimates!

The space-time volume is computed as

$$\hat{\mathbf{E}} = \mathbf{D}\hat{\boldsymbol{\alpha}}$$

K-SVD

Objective function: $\min_{D,X}\{\|Y - DX\|_F^2\}$ subject to $\forall i\ , \|x_i\|_0 \leq T_0$

where *Y* is the *observed data*, *D* is *dictionary* to be learnt and *X* is a T_0 *sparse vector.*

Alternating minimization is used as follows:

1. Keeping dictionary fixed, find the sparse representations using OMP.
2. Using these sparse representations, update one column at a time:

   Find SVD of the error matrix corresponding to the data-points that have non-zero coefficient corresponding to the current column.

   Replace dictionary column by first left singular vector, update corresponding coefficients by first right singular vector scaled by first singular value.

K-SVD

Objective function: $\min_{D,X}\{\|Y - DX\|_F^2\}$ subject to $\forall i\ , \|x_i\|_0 \leq T_0$

where *Y* is the *observed data*, *D* is *dictionary* to be learnt and *X* is a T_0 *sparse vector.*

Alternating minimization is used as follows:

1. Keeping dictionary fixed, find the sparse representations using OMP.
2. Using these sparse representations, update one column at a time:

   Find SVD of the error matrix corresponding to the data-points that have non-zero coefficient corresponding to the current column.

   Replace dictionary column by first left singular vector, update corresponding coefficients by first right singular vector scaled by first singular value.

K-SVD

Objective function: $\min_{D,X}\{\|Y - DX\|_F^2\}$ subject to $\forall i\ , \|x_i\|_0 \leq T_0$

where *Y* is the *observed data*, *D* is *dictionary* to be learnt and *X* is a T_0 *sparse vector.*

Alternating minimization is used as follows:

1. Keeping dictionary fixed, find the sparse representations using OMP.
2. Using these sparse representations, update one column at a time:

   Find SVD of the error matrix corresponding to the data-points that have non-zero coefficient corresponding to the current column.

   Replace dictionary column by first left singular vector, update corresponding coefficients by first right singular vector scaled by first singular value.

K-SVD

Objective function: $\min_{D,X}\{\|Y - DX\|_F^2\}$ subject to $\forall i\ , \|x_i\|_0 \leq T_0$

where *Y* is the *observed data*, *D* is *dictionary* to be learnt and *X* is a T_0 *sparse vector.*

Alternating minimization is used as follows:

1. Keeping dictionary fixed, find the sparse representations using OMP.
2. Using these sparse representations, update one column at a time:

   Find SVD of the error matrix excluding contribution from chosen column corresponding to the data-points that have non-zero coefficient corresponding to the current column.

   Replace dictionary column by first left singular vector, update corresponding coefficients by first right singular vector scaled by first singular value.

Constraints in the current system

- Maximum temporal resolution of the over-complete dictionary has to be pre-determined. To reconstruct videos at different temporal resolutions, we have to train different dictionaries.
- The hardware setup requires precise alignment of the camera. Artifacts due to imperfect alignment.
- Both dictionary learning and video reconstruction require a lot of time. Not suitable for real time applications.

Constraints in the current system

- Maximum temporal resolution of the over-complete dictionary has to be pre-determined. To reconstruct videos at different temporal resolutions, we have to train different dictionaries.
- The hardware setup requires precise alignment of the camera. Artifacts due to imperfect alignment.
- Both dictionary learning and video reconstruction require a lot of time. Not suitable for real time applications.

Constraints in the current system

- Maximum temporal resolution of the over-complete dictionary has to be pre-determined. To reconstruct videos at different temporal resolutions, we have to train different dictionaries.
- The hardware setup requires precise alignment of the camera. Artifacts due to imperfect alignment.
- Both dictionary learning and video reconstruction require a lot of time. Not suitable for real time applications.

List of Experiments

Observe the effect of following parameters on the reconstruction error

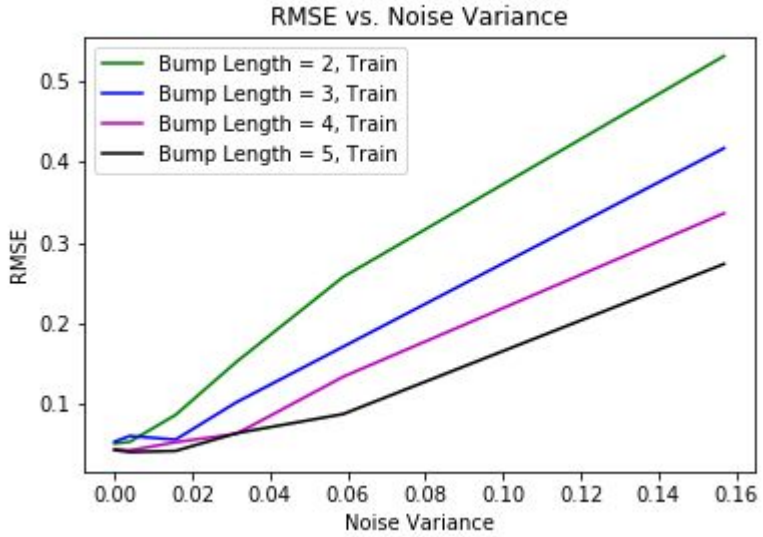
- Bump length
- Noise in the coded image
- Assumed sparsity of the videos in the dictionary basis
- No. of elements on the dictionary
- Patch size
- Stride
- Different sampling schemes

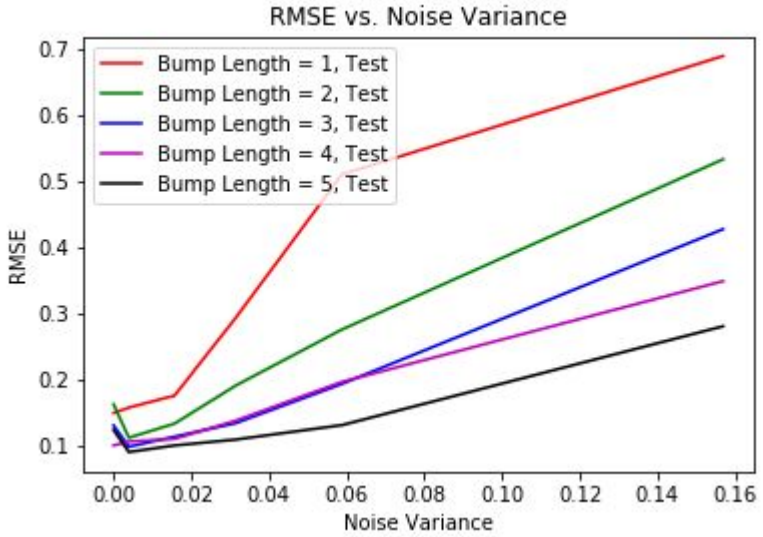
Details of Experiments

For each experiment, all but few (one or two) parameters are fixed:

- Temporal depth = 36
- Image height = 160
- Image width = 320
- Sparsity = 40
- Number of Videos = 20
- Bump Length = 3
- Number of basis per video segment = 625
- Patch size = 8
- Stride = 4

Effect of noise variance and bump length

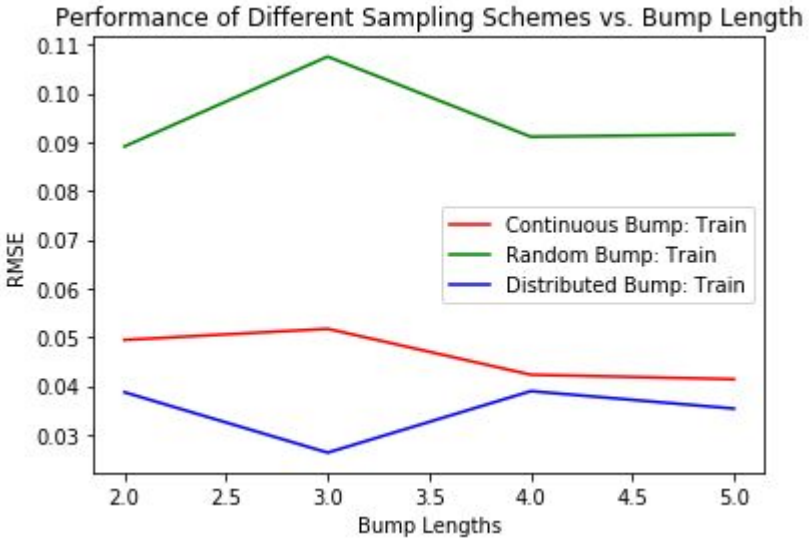


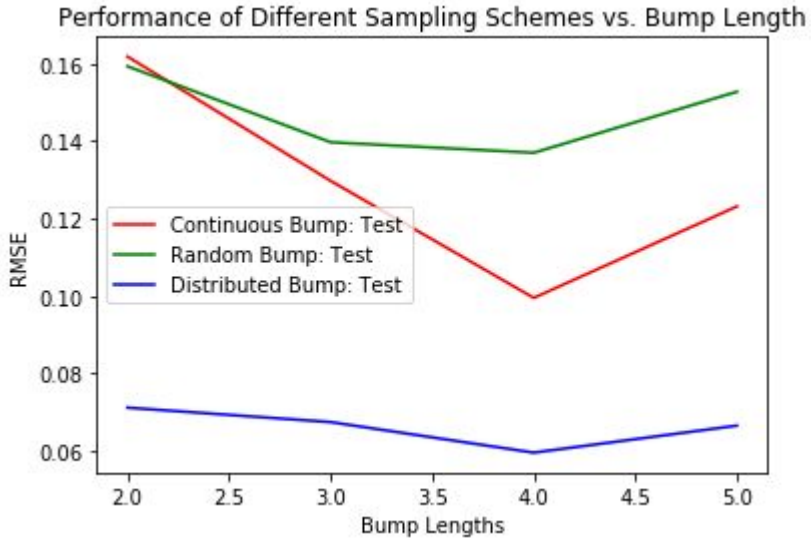


As bump length is increased from 1 to 5, reconstruction gets better. After a point increase in bump length (towards S(x,y,t)=1) is expected to increase RMSE.

As noise variance is increased RMSE increases in almost linear fashion.

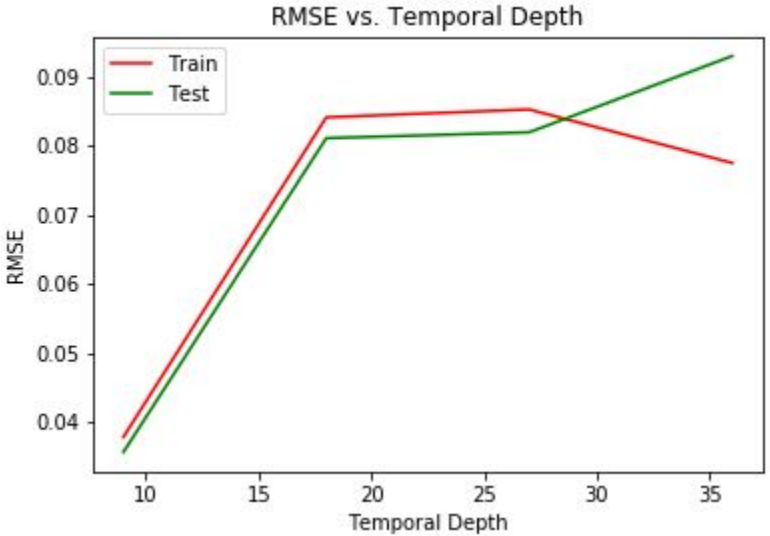
Effect of different sampling schemes





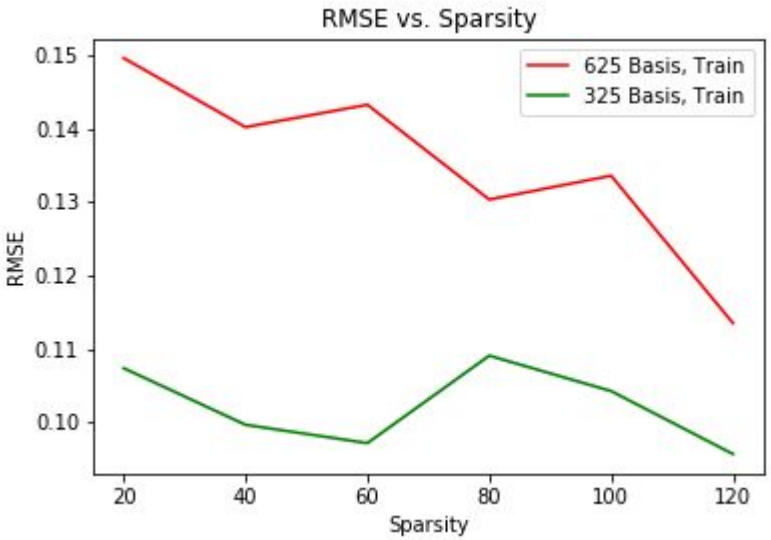
- *Continuous bump:* as per the hardware restrictions
- *Random sampling:* worst performance, as some spatial location may not be captured at all
- *Distributed bump:* Random within spatial location (continuity of bump relaxed) gives best RMSE

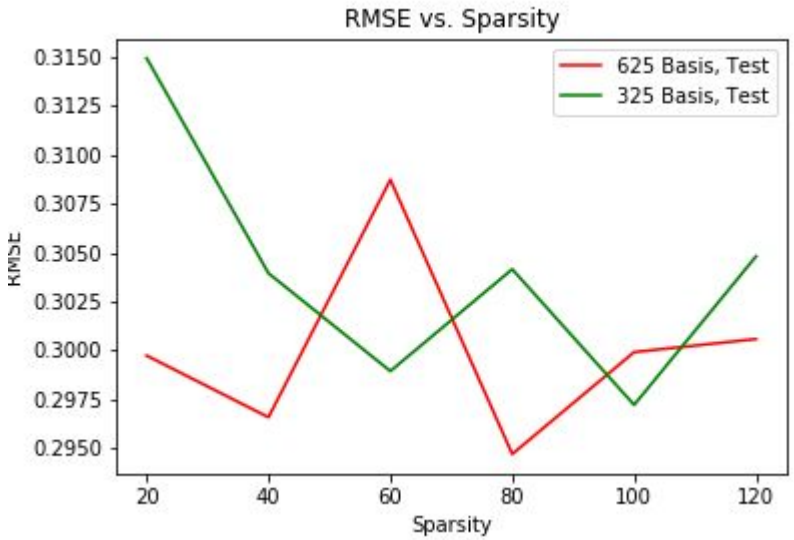
Effect of temporal depth



RMSE increases with temporal depth as expected since the number of elements to be recovered increases with same amount of evidence.

Effect of sparsity

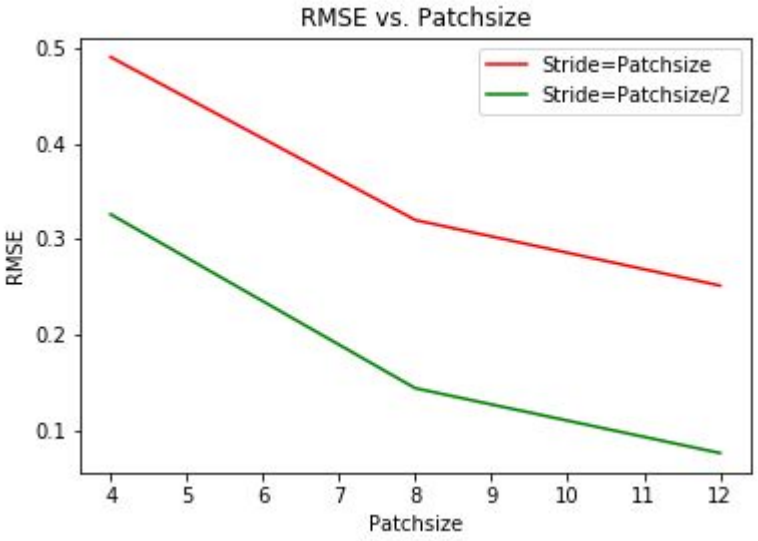




325 basis per video segment are observed to produce better reconstruction on the training set. For test, the results are chaotic.

Increasing sparsity reduces RMSE as expected.

Effect of patch size and stride



- Decreasing stride decreased the RMSE because of more overlap between neighbouring patches.
- Increasing patch size decreases the RMSE because each patch captures more information.
- Trend of RMSE with patch size is expected to saturate unless the number of basis in also increased.

Conclusions

- The proposed method can reconstruct the videos with high temporal resolution without compromising on spatial resolution. But artifacts are seen.

Conclusions

- The proposed method can reconstruct the videos with high temporal resolution without compromising on spatial resolution. But artifacts are seen.
- Effect of noise is as expected. Increasing bump length results in better reconstruction when bump lengths are small, but an optimal bump length less than 36 is expected.

Conclusions

- The proposed method can reconstruct the videos with high temporal resolution without compromising on spatial resolution. But artifacts are seen.
- Effect of noise is as expected. Increasing bump length results in better reconstruction when bump lengths are small, but an optimal bump length less than 36 is expected.
- Distributed bump sampling produces best results, but this has to be at the cost of increased hardware complexity (randomness is cool, provided each spatial location in captured in the coded image).

Conclusions

- The proposed method can reconstruct the videos with high temporal resolution without compromising on spatial resolution. But artifacts are seen.
- Effect of noise is as expected. Increasing bump length results in better reconstruction when bump lengths are small, but an optimal bump length less than 36 is expected.
- Distributed bump sampling produces best results, but this has to be at the cost of increased hardware complexity (randomness is cool, provided each spatial location in captured in the coded image).
- Increase in RMSE with temporal depth is as expected as we are trying to recover larger spatio-temporal volume from fixed number of measurements.

Conclusions

- 325 basis per videos were preferred than 625 videos. This is a surprising observation since the paper report using even bigger dictionary.

Conclusions

- 325 basis per videos were preferred than 625 videos. This is a surprising observation since the paper report using even bigger dictionary.
- Increasing sparsity upto 120 results in better video reconstruction.

Conclusions

- 325 basis per videos were preferred than 625 videos. This is a surprising observation since the paper report using even bigger dictionary.
- Increasing sparsity upto 120 results in better video reconstruction.
- Increasing patch size helps capture more information in the basis, decreasing the RMSE.

Conclusions

- 325 basis per videos were preferred than 625 videos. This is a surprising observation since the paper report using even bigger dictionary.
- Increasing sparsity upto 120 results in better video reconstruction.
- Increasing patch size helps capture more information in the basis, increasing the RMSE.
- Reducing stride makes patches overlapping. Thus, the artifacts are reduced and reconstruction is better.

THE END