Class-specific Image Denoising using Importance Sampling

Milad Niknejad, José M. Bioucas-Dias, Mário A. T. Figueiredo

Instituto de Telecomunicações, Instituto Superior Técnico,
University of Lisbon, Portugal

SpaRTaN
Sparse Representations and Compressed
Sensing Training Network

Overview

Class-specific image denoising

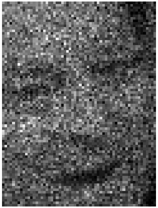
Often, the image to be denoised belongs to a known specific class,

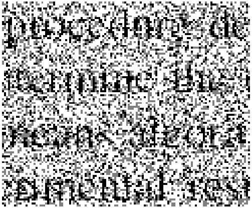
Examples: text/document, face, fingerprint, a specific type of medical image (*e.g.*, brain MRI), ...

Class-specific image denoising

Often, the image to be denoised belongs to a known specific class,

Examples: text/document, face, fingerprint, a specific type of medical image (*e.g.*, brain MRI), ...

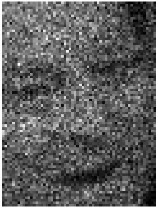


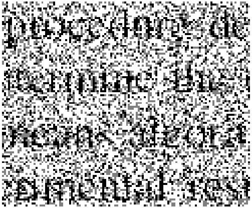


Class-specific image denoising

Often, the image to be denoised belongs to a known specific class,

Examples: text/document, face, fingerprint, a specific type of medical image (*e.g.*, brain MRI), ...





This knowledge should be exploited by the denoising method!

Class-specific image denoising

Assumption: A dataset of clean images of the same class is available.

Patch-based image denoising

We work with patches (in both noisy and external dataset images).

Patch-based image denoising

We work with patches (in both noisy and external dataset images).
Gaussian noise observation model:

$$\mathbf{y}_i = \mathbf{x}_i + \mathbf{v}_i$$

$\mathbf{x}_i$ is a patch of the original image; $\mathbf{y}_i$ is the corresponding noisy patch; $\mathbf{v}_i$ is i.i.d. Gaussian noise (variance σ^2).

Patch-based image denoising

We work with patches (in both noisy and external dataset images).
Gaussian noise observation model:

$$\mathbf{y}_i = \mathbf{x}_i + \mathbf{v}_i$$

$\mathbf{x}_i$ is a patch of the original image; $\mathbf{y}_i$ is the corresponding noisy patch;
$\mathbf{v}_i$ is i.i.d. Gaussian noise (variance σ^2).

Poisson noise observation model:

$$\mathbf{y}_{i,j} \sim \mathcal{P}(\mathbf{x}_{i,j}).$$

$\mathbf{x}_{i,j}$ is the j^{th} pixel of $\mathbf{x}_i$.
$\mathcal{P}$ is a Poisson distribution with mean $\mathbf{x}_{i,j}$.

Patch-based image denoising

We work with patches (in both noisy and external dataset images).
Gaussian noise observation model:

$$\mathbf{y}_i = \mathbf{x}_i + \mathbf{v}_i$$

$\mathbf{x}_i$ is a patch of the original image; $\mathbf{y}_i$ is the corresponding noisy patch;
$\mathbf{v}_i$ is i.i.d. Gaussian noise (variance σ^2).

Poisson noise observation model:

$$\mathbf{y}_{i,j} \sim \mathcal{P}(\mathbf{x}_{i,j}).$$

$\mathbf{x}_{i,j}$ is the j^{th} pixel of $\mathbf{x}_i$.
$\mathcal{P}$ is a Poisson distribution with mean $\mathbf{x}_{i,j}$.

Goal: recover the clean patch $\mathbf{x}_i$ from the noisy one $\mathbf{y}_i$

MMSE (*minimum mean squared error*) estimation

MMSE patch estimate $(p(\mathbf{y} = \mathbf{y}_i) : p(\mathbf{y}_i))$:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\; p(\mathbf{x}|\mathbf{y}_i)\; d\mathbf{x} = \int_{\mathbb{R}^p} \mathbf{x}\; \frac{p(\mathbf{y}_i|\mathbf{x})\, p(\mathbf{x})}{p(\mathbf{y}_i)}\; d\mathbf{x}$$

MMSE (*minimum mean squared error*) estimation

MMSE patch estimate ($p(\mathbf{y} = \mathbf{y}_i) : p(\mathbf{y}_i)$):

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\; p(\mathbf{x}|\mathbf{y}_i)\; d\mathbf{x} = \int_{\mathbb{R}^p} \mathbf{x}\; \frac{p(\mathbf{y}_i|\mathbf{x})\, p(\mathbf{x})}{p(\mathbf{y}_i)}\; d\mathbf{x}$$

- This multi-dimensional integral is intractable, in general (exception: Gaussian noise and Gaussian prior).

MMSE (*minimum mean squared error*) estimation

MMSE patch estimate ($p(\mathbf{y} = \mathbf{y}_i) : p(\mathbf{y}_i)$):

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\; p(\mathbf{x}|\mathbf{y}_i)\; d\mathbf{x} = \int_{\mathbb{R}^p} \mathbf{x}\; \frac{p(\mathbf{y}_i|\mathbf{x})\, p(\mathbf{x})}{p(\mathbf{y}_i)}\; d\mathbf{x}$$

- This multi-dimensional integral is intractable, in general (exception: Gaussian noise and Gaussian prior).
- Monte-Carlo approximation: obtain samples $\mathbf{x}_j$ from $p(\mathbf{x}|\mathbf{y}_i)$

$$\hat{\hat{\mathbf{x}}}_i = \frac{1}{n}\sum_{j=1}^{n} \mathbf{x}_j \qquad \lim_{n\to\infty} \hat{\hat{\mathbf{x}}}_i = \hat{\mathbf{x}}_i$$

MMSE (*minimum mean squared error*) estimation

MMSE patch estimate $(p(\mathbf{y} = \mathbf{y}_i) : p(\mathbf{y}_i))$:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\, p(\mathbf{x}|\mathbf{y}_i)\, d\mathbf{x} = \int_{\mathbb{R}^p} \mathbf{x}\, \frac{p(\mathbf{y}_i|\mathbf{x})\, p(\mathbf{x})}{p(\mathbf{y}_i)}\, d\mathbf{x}$$

- This multi-dimensional integral is intractable, in general (exception: Gaussian noise and Gaussian prior).
- Monte-Carlo approximation: obtain samples $\mathbf{x}_j$ from $p(\mathbf{x}|\mathbf{y}_i)$

$$\hat{\hat{\mathbf{x}}}_i = \frac{1}{n}\sum_{j=1}^{n} \mathbf{x}_j \qquad \lim_{n\to\infty} \hat{\hat{\mathbf{x}}}_i = \hat{\mathbf{x}}_i$$

- However, sampling from $p(\mathbf{x}|\mathbf{y}_i)$ is also intractable.

MMSE (*minimum mean squared error*) estimation

MMSE patch estimate ($p(\mathbf{y} = \mathbf{y}_i) : p(\mathbf{y}_i)$):

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\, p(\mathbf{x}|\mathbf{y}_i)\, d\mathbf{x} = \int_{\mathbb{R}^p} \mathbf{x}\, \frac{p(\mathbf{y}_i|\mathbf{x})\, p(\mathbf{x})}{p(\mathbf{y}_i)}\, d\mathbf{x}$$

- This multi-dimensional integral is intractable, in general (exception: Gaussian noise and Gaussian prior).
- Monte-Carlo approximation: obtain samples $\mathbf{x}_j$ from $p(\mathbf{x}|\mathbf{y}_i)$

$$\hat{\hat{\mathbf{x}}}_i = \frac{1}{n}\sum_{j=1}^{n} \mathbf{x}_j \qquad\qquad \lim_{n\to\infty} \hat{\hat{\mathbf{x}}}_i = \hat{\mathbf{x}}_i$$

- However, sampling from $p(\mathbf{x}|\mathbf{y}_i)$ is also intractable.
- Can we approximate $\hat{\mathbf{x}}_i$ by sampling from another distribution?

Self-Normalized Importance Sampling (SNIS)

Goal: to compute (or approximate)

$$\mathbb{E}[f(\mathbf{z})] = \int f(\mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}.$$

Self-Normalized Importance Sampling (SNIS)

Goal: to compute (or approximate)

$$\mathbb{E}[f(\mathbf{z})] = \int f(\mathbf{z})\, p(\mathbf{z})\; d\mathbf{z}.$$

- Let $\tilde{p}(\mathbf{z}) = c\, p(\mathbf{z})$ be an un-normalized version of $p(\mathbf{z})$.

Self-Normalized Importance Sampling (SNIS)

Goal: to compute (or approximate)

$$\mathbb{E}[f(\mathbf{z})] = \int f(\mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}.$$

- Let $\tilde{p}(\mathbf{z}) = c\, p(\mathbf{z})$ be an un-normalized version of $p(\mathbf{z})$.
- Let $\tilde{q}(\mathbf{z}) = b\, q(\mathbf{z})$ be another un-normalized density; assume it is possible/easy to obtain samples $\mathbf{z}_1, ..., \mathbf{z}_n \sim q(\mathbf{z})$.
- Constants c and b may be unknown.

Self-Normalized Importance Sampling (SNIS)

Goal: to compute (or approximate)

$$\mathbb{E}[f(\mathbf{z})] = \int f(\mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}.$$

- Let $\tilde{p}(\mathbf{z}) = c\, p(\mathbf{z})$ be an un-normalized version of $p(\mathbf{z})$.
- Let $\tilde{q}(\mathbf{z}) = b\, q(\mathbf{z})$ be another un-normalized density; assume it is possible/easy to obtain samples $\mathbf{z}_1, ..., \mathbf{z}_n \sim q(\mathbf{z})$.
- Constants c and b may be unknown.

$$\hat{\mathbb{E}}_n[f(\mathbf{z})] = \frac{\sum_{j=1}^{n} f(\mathbf{z}_j) w(\mathbf{z}_j)}{\sum_{j=1}^{n} w(\mathbf{z}_j)}, \qquad w(\mathbf{z}_j) = \frac{\tilde{p}(\mathbf{z}_j)}{\tilde{q}(\mathbf{z}_j)}.$$

Self-Normalized Importance Sampling (SNIS)

Goal: to compute (or approximate)

$$\mathbb{E}[f(\mathbf{z})] = \int f(\mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}.$$

- Let $\tilde{p}(\mathbf{z}) = c\, p(\mathbf{z})$ be an un-normalized version of $p(\mathbf{z})$.
- Let $\tilde{q}(\mathbf{z}) = b\, q(\mathbf{z})$ be another un-normalized density; assume it is possible/easy to obtain samples $\mathbf{z}_1, ..., \mathbf{z}_n \sim q(\mathbf{z})$.
- Constants c and b may be unknown.

$$\hat{\mathbb{E}}_n[f(\mathbf{z})] = \frac{\sum_{j=1}^{n} f(\mathbf{z}_j) w(\mathbf{z}_j)}{\sum_{j=1}^{n} w(\mathbf{z}_j)}, \qquad w(\mathbf{z}_j) = \frac{\tilde{p}(\mathbf{z}_j)}{\tilde{q}(\mathbf{z}_j)}.$$

- As in plain Monte-Carlo: $\lim_{n\to\infty} \hat{\mathbb{E}}_n[f(\mathbf{z})] = \mathbb{E}[f(\mathbf{z})]$

Applying SNIS for MMSE patch estimation

Back to our problem:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\ p(\mathbf{x}|\mathbf{y}_i)\ d\mathbf{x}.$$

Applying SNIS for MMSE patch estimation

Back to our problem:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\; p(\mathbf{x}|\mathbf{y}_i)\; d\mathbf{x}.$$

- Instead of sampling from $p(\mathbf{x}|\mathbf{y}_i)$, use samples $\mathbf{x}_1, ..., \mathbf{x}_n$ from $p(\mathbf{x})$;

Applying SNIS for MMSE patch estimation

Back to our problem:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\; p(\mathbf{x}|\mathbf{y}_i)\; d\mathbf{x}.$$

- Instead of sampling from $p(\mathbf{x}|\mathbf{y}_i)$, use samples $\mathbf{x}_1, ..., \mathbf{x}_n$ from $p(\mathbf{x})$;
- Simply use samples from the external dataset of clean patches.

Applying SNIS for MMSE patch estimation

Back to our problem:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\ p(\mathbf{x}|\mathbf{y}_i)\ d\mathbf{x}.$$

- Instead of sampling from $p(\mathbf{x}|\mathbf{y}_i)$, use samples $\mathbf{x}_1, ..., \mathbf{x}_n$ from $p(\mathbf{x})$;
- Simply use samples from the external dataset of clean patches.
- Use these samples in SNIS

$$\hat{\hat{\mathbf{x}}}_i = \hat{\mathbb{E}}_n[\mathbf{x}|\mathbf{y}_i] = \frac{\sum_{j=1}^{n} \mathbf{x}_j\ w_j}{\sum_{j=1}^{n} w_j}, \qquad w_j = p(\mathbf{y}_i|\mathbf{x} = \mathbf{x}_j)$$

Applying SNIS for MMSE patch estimation

Back to our problem:

$$\hat{\mathbf{x}}_i = \mathbb{E}[\mathbf{x}|\mathbf{y}_i] = \int_{\mathbb{R}^p} \mathbf{x}\; p(\mathbf{x}|\mathbf{y}_i)\; d\mathbf{x}.$$

- Instead of sampling from $p(\mathbf{x}|\mathbf{y}_i)$, use samples $\mathbf{x}_1, ..., \mathbf{x}_n$ from $p(\mathbf{x})$;
- Simply use samples from the external dataset of clean patches.
- Use these samples in SNIS

$$\hat{\hat{\mathbf{x}}}_i = \hat{\mathbb{E}}_n[\mathbf{x}|\mathbf{y}_i] = \frac{\sum_{j=1}^{n} \mathbf{x}_j\, w_j}{\sum_{j=1}^{n} w_j}, \qquad w_j = p(\mathbf{y}_i|\mathbf{x} = \mathbf{x}_j)$$

- Why? $\tilde{p}(\mathbf{x}) = p(\mathbf{y}|\mathbf{x})\, p(\mathbf{x})$, $c = 1/p(\mathbf{y})$ and $\tilde{q}(\mathbf{x}) = p(\mathbf{x})$.

Applying SNIS for MMSE patch estimation (II)

- For Gaussian noise, the weights are simply

$$w_j = \exp\left[-\frac{\|\mathbf{x}_j - \mathbf{y}_i\|_2^2}{2\sigma^2}\right]$$

Applying SNIS for MMSE patch estimation (II)

- For Gaussian noise, the weights are simply

$$w_j = \exp\left[-\frac{\|\mathbf{x}_j - \mathbf{y}_i\|_2^2}{2\sigma^2}\right]$$

- This formulation provides a SNIS interpretation to the (external) non-local means (NLM) approach [Zontak et al., 2012].

Applying SNIS for MMSE patch estimation (II)

- For Gaussian noise, the weights are simply

$$w_j = \exp\left[-\frac{\|\mathbf{x}_j - \mathbf{y}_i\|_2^2}{2\sigma^2}\right]$$

- This formulation provides a SNIS interpretation to the (external) non-local means (NLM) approach [Zontak et al., 2012].

- It can be adapted to other image restoration tasks.

Applying SNIS for MMSE patch estimation (II)

- For Gaussian noise, the weights are simply

$$w_j = \exp\left[-\frac{\|\mathbf{x}_j - \mathbf{y}_i\|_2^2}{2\sigma^2}\right]$$

- This formulation provides a SNIS interpretation to the (external) non-local means (NLM) approach [Zontak et al., 2012].
- It can be adapted to other image restoration tasks.
- It can be generalized to other noise models (*e.g.*, Poisson: next talk):

Making it work: proposed approach

Key observations:

Making it work: proposed approach

Key observations:

1. Using samples from $p(\mathbf{x})$ is sub-optimal, as it may have high variance (or even infinite variance). It requires very large n.

Making it work: proposed approach

Key observations:

1. Using samples from $p(\mathbf{x})$ is sub-optimal, as it may have high variance (or even infinite variance). It requires very large n.

2. The proposal distribution should be made similar to each target distribution $p(\mathbf{x}|\mathbf{y}_i)$: Estimator with lower MMSE for limited number of samples [Koblents et al., 2012].

Making it work: proposed approach

Key observations:

1. Using samples from $p(\mathbf{x})$ is sub-optimal, as it may have high variance (or even infinite variance). It requires very large n.
2. The proposal distribution should be made similar to each target distribution $p(\mathbf{x}|\mathbf{y}_i)$: Estimator with lower MMSE for limited number of samples [Koblents et al., 2012].

Proposed approach:

1. Cluster the patches in the external dataset.

Making it work: proposed approach

Key observations:

1. Using samples from $p(\mathbf{x})$ is sub-optimal, as it may have high variance (or even infinite variance). It requires very large n.
2. The proposal distribution should be made similar to each target distribution $p(\mathbf{x}|\mathbf{y}_i)$: Estimator with lower MMSE for limited number of samples [Koblents et al., 2012].

Proposed approach:

1. Cluster the patches in the external dataset.
2. Assign each noisy patch to the closest cluster.

Making it work: proposed approach

Key observations:

1. Using samples from $p(\mathbf{x})$ is sub-optimal, as it may have high variance (or even infinite variance). It requires very large n.
2. The proposal distribution should be made similar to each target distribution $p(\mathbf{x}|\mathbf{y}_i)$: Estimator with lower MMSE for limited number of samples [Koblents et al., 2012].

Proposed approach:

1. Cluster the patches in the external dataset.
2. Assign each noisy patch to the closest cluster.
3. Use the corresponding clean patches as samples from the proposal distribution for SNIS.

More details

Clustering:

- The dataset of clean patches is clustered into K clusters.
- We use classification-EM (CEM) with Gaussian clusters.

More details

Clustering:

- The dataset of clean patches is clustered into K clusters.

- We use classification-EM (CEM) with Gaussian clusters.
  - Input: clean patches in the external dataset.
  - Output: set of K multivariate Gaussians $\mathcal{N}_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ and K clusters of clean patches.

More details

Clustering:

- The dataset of clean patches is clustered into K clusters.

- We use classification-EM (CEM) with Gaussian clusters.
  - Input: clean patches in the external dataset.
  - Output: set of K multivariate Gaussians $\mathcal{N}_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ and K clusters of clean patches.

- Note: each multivariate Gaussian is an approximation of the clean patch distribution in each cluster.

More details

Clustering:

- The dataset of clean patches is clustered into K clusters.
- We use classification-EM (CEM) with Gaussian clusters.
  - Input: clean patches in the external dataset.
  - Output: set of K multivariate Gaussians $\mathcal{N}_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ and K clusters of clean patches.
- Note: each multivariate Gaussian is an approximation of the clean patch distribution in each cluster.

Classification of each noisy patch:

- $\widehat{k} = \arg\max_k p_Y(\mathbf{y}|k)$, where $p_Y(\cdot|k) \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k + \sigma^2 \mathbf{I})$.

Some Comments

- The weighted patch average from external dataset has been used before [Levin et al., 2012], [Zontak et al., 2012].

Some Comments

- The weighted patch average from external dataset has been used before [Levin et al., 2012], [Zontak et al., 2012].
- The importance sampling (IS) viewpoint seems to be new.

Some Comments

- The weighted patch average from external dataset has been used before [Levin et al., 2012], [Zontak et al., 2012].
- The importance sampling (IS) viewpoint seems to be new.
- This viewpoint allows resorting to the vast literature on IS.

Some Comments

- The weighted patch average from external dataset has been used before [Levin et al., 2012], [Zontak et al., 2012].
- The importance sampling (IS) viewpoint seems to be new.
- This viewpoint allows resorting to the vast literature on IS.
- In our experiments, we use a proposal in [Koblents et al., 2012], which applies a hard threshold on the importance weights:

$$w_j \leftarrow \mathcal{T}_h(w_j). \tag{1}$$

Some Comments

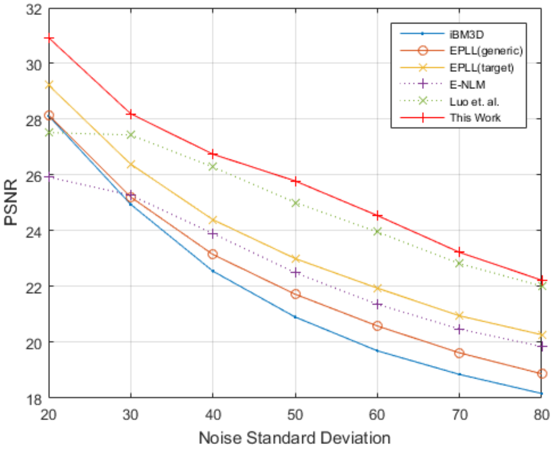
- The weighted patch average from external dataset has been used before [Levin et al., 2012], [Zontak et al., 2012].
- The importance sampling (IS) viewpoint seems to be new.
- This viewpoint allows resorting to the vast literature on IS.
- In our experiments, we use a proposal in [Koblents et al., 2012], which applies a hard threshold on the importance weights:

$$w_j \leftarrow \mathcal{T}_h(w_j). \quad (1)$$

This technique reduces the variance of the estimator.

Experiments: Face Images

Results on the Gore face dataset.



Examples

Noisy ($\sigma = 30$)

BM3D(PSNR=29.46)

EPLL (PSNR=28.97)

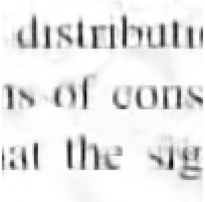
Class-spesific EPLL
(PSNR=29.91)

[Luo et al., 2015]
(PSNR=32.20)

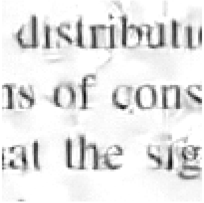
This work
(PSNR=33.02)

Experiments: Text Images

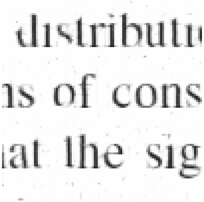
Noisy ($\sigma = 50$)



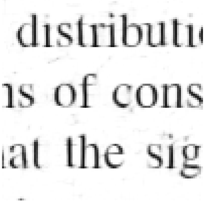
BM3D(PSNR=20.14)



Cl-specific EPLL
(PSNR=14.55)



[Luo et al., 2015]
(PSNR=23.92)



This work
(PSNR=25.44)

Conclusions

- By selecting class-specific external datasets, it is possible to adapt to specific image classes, outperforming generic denoisers.

Conclusions

- By selecting class-specific external datasets, it is possible to adapt to specific image classes, outperforming generic denoisers.
- MMSE estimation for inverse problem can be approximated by using importance sampling.

Conclusions

- By selecting class-specific external datasets, it is possible to adapt to specific image classes, outperforming generic denoisers.
- MMSE estimation for inverse problem can be approximated by using importance sampling.
- This viewpoint allows using the many methods proposed to improve importance sampling.

Conclusions

- By selecting class-specific external datasets, it is possible to adapt to specific image classes, outperforming generic denoisers.
- MMSE estimation for inverse problem can be approximated by using importance sampling.
- This viewpoint allows using the many methods proposed to improve importance sampling.
- Sampling from the external dataset can be used to approximate MMSE estimation under true underlying distribution of clean patches.

Conclusions

- By selecting class-specific external datasets, it is possible to adapt to specific image classes, outperforming generic denoisers.
- MMSE estimation for inverse problem can be approximated by using importance sampling.
- This viewpoint allows using the many methods proposed to improve importance sampling.
- Sampling from the external dataset can be used to approximate MMSE estimation under true underlying distribution of clean patches.
- Our method does not require to fit any parametric distribution on dataset.

References

E. Luo, S. H Chan, and T. Q Nguyen,
Adaptive image denoising by targeted databases
IEEE Transactions on Image Processing, vol. 24, pp. 2167–2181, 2015.

A. Levin, B. Nadler, F. Durand, and W. T. Freeman,
Patch complexity, finite pixel correlations and optimal denoising.
European Conference on Computer Vision, pp. 73–86, 2012.

M. Zontak and M. Irani,
Internal statistics of a single natural image.
IEEE Conference on Computer Vision and Pattern Recognition, pp. 977–984, 2012.

E. Koblents and J.Miguez,
A population Monte–Carlo scheme with transformed weights and its application to stochastic kinetic models
Statistics and Computing, vol. 25, pp. 407–425, 2015.

Thanks for your attention

Next talk: applying importance sampling to Poisson noise