Assignment 5: CS 754

Krushnakant Bhattad	Devansh Jain
190100036	190100044

April 19, 2021

Contents

Question 1

Part (2)

Question: In Eqn. (6), which terms are obtained from the prior and which terms are obtained from the likelihood? What is the prior used in the paper? What is the likelihood used in the paper?

Answer:

The Terms

Let's break down the function as:

$$\text{Expression 1: } \sum_{i,k} \rho(f_{i,k} \cdot I_1) + \rho(f_{i,k} \cdot (I - I_1))$$

$$\text{Expression 2: } \lambda \sum_{i \in S_1, k} \rho(f_{i,k} \cdot I_1 - f_{i,k} \cdot I) + \lambda \sum_{i \in S_2, k} \rho(f_{i,k} \cdot I_1)$$

Then, we note that the terms in Expression 1 have come from our prior belief, and that the terms in expression 2 have come from the likelihood term.

The Prior used

The Prior used in the paper is the sparsity prior.

A sparse distribution can be achieved by mixing only two laplacian distributions: a narrow distribution centered on zero and a broad distribution centered on zero.

For example,

$$\Pr(x) = \frac{\pi_1}{2s_1} e^{-|x|/s_1} + \frac{\pi_2}{2s_2} e^{-|x|/s_2}$$

where learned values found are $s_1 = 0.01, s_2 = 0.05, \pi_1 = 0.4, \pi_2 = 0.6$.

We define a distribution over images by assuming that derivative filters are independent over space and orientation so that our prior over images is given by:

$$\Pr(I) \approx \prod_{i,k} \Pr(f_{i,k} \cdot I)$$

where $f \cdot I$ denotes the inner product between a linear filter f and an image I , and $f_{i,k}$ is the k^{th} derivative filter centered on pixel i . We enforce this prior using negative log of the probability in the cost function.

The Likelihood used

The likelihood used in the paper depends upon two things: first, the given input image I.

Next, the two sets of image locations S_1, S_2 so that gradients in location S_1 belong to layer 1 and gradients in location S_2 belong to layer 2 .

We enforce the agreement with the labeled gradients using the terms in expression 2.

$\sum_{i \in S_1, k} \rho(f_{i,k} \cdot I_1 - f_{i,k} \cdot I)$ enforces that gradients in location S_1 belong to layer 1, and

$\sum_{i \in S_2, k} \rho(f_{i,k} \cdot I_1)$ enforces that gradients in location S_2 belong to layer 2 (here we use $I - I_2 = I_1$)

Part (3)

Question:

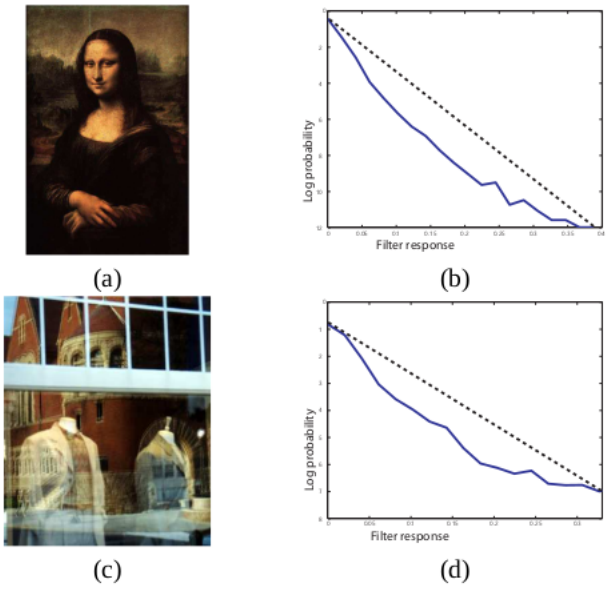
Why does the paper use a likelihood term that is different from the Gaussian prior?

Answer:

A remarkably robust property of natural images that has received much attention lately is the fact that when derivative filters are applied to natural images, the filter outputs tend to be sparse. The log-histograms of derivative filters lie below the straight line connecting the minimal and maximal values. We refer to such distributions as sparse.

For example, it has been illustrated in literature that the histogram of the vertical derivative filter is peaked at zero and falls off much faster than a Gaussian.

The following figure shows a natural image on the left side, and on the right side, a plot of d_y derivative filter output in blue; and a dashed straight line connecting the minimal and maximal values. This supports what we have said so far.



When we look at the logarithm of the histogram the curve is always below the straight line connecting the maximum and minimum values.

This should be contrasted with the Gaussian distribution (that is always above the straight line) or the Laplacian distribution (that is simply a straight line in the log domain)

It was shown (in literature) that the fact that the log distribution is always below the straight line, is crucial for obtaining transparency decomposition from a single image.

Distributions that are above the straight line (like the gaussian) will prefer to split an edge of unit contrast into two edges (one in each layer) with half the contrast, but what we want, is achieved by distributions below the line: They will prefer decompositions in which the edge only appears in one of the layers but not in the other. (that is, treat the images I_1, and I_2 as separate and not mixed unlike the gaussian distribution).
That is why we use the sparse distribution, which was defined in the previous part in the prior section.

Question 2

Instructions for running the code:

- After extracting submitted file, look for a directory named `q2`, and `cd` (change directory) to it.
- Run the file `q2.m`.
- The plots can be found in `./plots/`.

Deriving expression for maximum a posteriori (MAP) Estimate

Given:

$$\begin{aligned}
&\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\eta} \\
&\boldsymbol{y} \in \mathbb{R}^m \\
&\boldsymbol{\Phi} \in \mathbb{R}^{m \times n} \ (m \ll n) \\
&\boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{x} \sim \mathcal{N}(0, \boldsymbol{\Sigma}) \\
&\boldsymbol{\eta} \in \mathbb{R}^m, \boldsymbol{\eta} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I}_{m \times m})
\end{aligned}$$

MAP of $\boldsymbol{x}$:

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\phi}) \cdot p(\boldsymbol{x})$$

(Noise $\boldsymbol{\eta} = \boldsymbol{y} - \boldsymbol{\phi x}$ is Gaussian modelled, so is the original vector $\boldsymbol{x}$)

$$= \arg\max_{\boldsymbol{x}} \left(\frac{1}{(\sigma^{2m})^{0.5}(2\pi)^{m/2}} \exp\left(-\frac{(\boldsymbol{y} - \boldsymbol{\phi x})^t (\sigma^2 \boldsymbol{I})^{-1} (\boldsymbol{y} - \boldsymbol{\phi x})}{2} \right) \right) \cdot \left(\frac{1}{(|\boldsymbol{\Sigma}|)^{0.5}(2\pi)^{n/2}} \exp\left(-\frac{(\boldsymbol{x})^t (\boldsymbol{\Sigma})^{-1} (\boldsymbol{x})}{2} \right) \right)$$

$$= \arg\max_{\boldsymbol{x}} \left(\frac{1}{(\sigma^{2m})^{0.5}(|\boldsymbol{\Sigma}|)^{0.5}(2\pi)^{(m+n)/2}} \right) \cdot \exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{\phi x}\|_2^2}{2\sigma^2} - \frac{(\boldsymbol{x})^t (\boldsymbol{\Sigma})^{-1} (\boldsymbol{x})}{2} \right)$$

$$= \arg\max_{\boldsymbol{x}} \ \exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{\phi x}\|_2^2}{2\sigma^2} - \frac{(\boldsymbol{x})^t (\boldsymbol{\Sigma})^{-1} (\boldsymbol{x})}{2} \right) \quad (\sigma, \Sigma \text{ are independent of x})$$

$$= \arg\min_{\boldsymbol{x}} \left(\frac{\|\boldsymbol{y} - \boldsymbol{\phi x}\|_2^2}{2\sigma^2} + \frac{(\boldsymbol{x})^t (\boldsymbol{\Sigma})^{-1} (\boldsymbol{x})}{2} \right) \quad (\arg\max_x f(x) = \arg\min_x -\log f(x))$$

$$= \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\phi}, \boldsymbol{\Sigma}, \sigma)$$

$$\left. \frac{\partial f(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\phi}, \boldsymbol{\Sigma}, \sigma)}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x} = \hat{\boldsymbol{x}}} = \frac{2(-\boldsymbol{\phi}^t)(\boldsymbol{y} - \boldsymbol{\phi}\hat{\boldsymbol{x}})}{2\sigma^2} + \frac{2\boldsymbol{\Sigma}^{-1}\hat{\boldsymbol{x}}}{2} = 0$$

$$\frac{\boldsymbol{\phi}^t \boldsymbol{y}}{\sigma^2} - \frac{\boldsymbol{\phi}^t \boldsymbol{\phi}\hat{\boldsymbol{x}}}{\sigma^2} - \boldsymbol{\Sigma}^{-1}\hat{\boldsymbol{x}} = 0$$

$$\frac{\boldsymbol{\phi}^t \boldsymbol{y}}{\sigma^2} = \left(\frac{\boldsymbol{\phi}^t \boldsymbol{\phi}}{\sigma^2} + \boldsymbol{\Sigma}^{-1} \right) \hat{\boldsymbol{x}}$$

$$\hat{\boldsymbol{x}} = \left(\frac{\boldsymbol{\phi}^t \boldsymbol{\phi}}{\sigma^2} + \boldsymbol{\Sigma}^{-1} \right)^{-1} \frac{\boldsymbol{\phi}^t \boldsymbol{y}}{\sigma^2}$$

Making the calculation computationally faster:

$$\begin{aligned}
\left(\frac{\boldsymbol{\phi}^t \boldsymbol{\phi}}{\sigma^2} + \boldsymbol{\Sigma}^{-1} \right)^{-1} &= \left(\boldsymbol{\Sigma}^{-1} + \boldsymbol{\phi}^t \left(\frac{\boldsymbol{I}_{m \times m}}{\sigma^2} \right) \boldsymbol{\phi} \right)^{-1} \\
&= \left(\boldsymbol{\Sigma} - \boldsymbol{\Sigma}\boldsymbol{\phi}^t \left((\sigma^2 \boldsymbol{I}_{m \times m}) + (\boldsymbol{\phi}\boldsymbol{\Sigma}\boldsymbol{\phi}^t) \right)^{-1} \boldsymbol{\phi}\boldsymbol{\Sigma} \right) \quad \text{(Using Woodbury matrix identity)}
\end{aligned}$$

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{\Sigma} - \boldsymbol{\Sigma}\boldsymbol{\phi}^t \left((\sigma^2 \boldsymbol{I}_{m\times m}) + (\boldsymbol{\phi}\boldsymbol{\Sigma}\boldsymbol{\phi}^t)\right)^{-1} \boldsymbol{\phi}\boldsymbol{\Sigma}\right) \frac{\boldsymbol{\phi}^t \boldsymbol{y}}{\sigma^2}$$

$$= \frac{\left(\boldsymbol{\Sigma}\boldsymbol{\phi}^t - \boldsymbol{\Sigma}\boldsymbol{\phi}^t \left((\sigma^2 \boldsymbol{I}_{m\times m}) + (\boldsymbol{\phi}\boldsymbol{\Sigma}\boldsymbol{\phi}^t)\right)^{-1} \boldsymbol{\phi}\boldsymbol{\Sigma}\boldsymbol{\phi}^t\right) \boldsymbol{y}}{\sigma^2}$$

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{C_1} - \boldsymbol{C_1} \left(\sigma^2 \boldsymbol{I}_{m\times m} + \boldsymbol{C_2}\right)^{-1} \boldsymbol{C_2}\right) \boldsymbol{y}/\sigma^2$$

where, $\boldsymbol{C_1} = \boldsymbol{\Sigma}\boldsymbol{\phi}^t$ and $\boldsymbol{C_2} = \boldsymbol{\phi}\boldsymbol{\Sigma}\boldsymbol{\phi}^t$

Explaining the Code

Generating original signal

1. Generating a random orthonormal matrix

```
U = zeros(n);
vi = randn(n,1);
U(:,1) = vi ./ norm(vi);
for i=2:n
  nrm = 0;
  while nrm < tol
    vi = randn(n,1);
    vi = vi - U(:,1:i-1) * (U(:,1:i-1)' * vi);
    nrm = norm(vi);
  end
  U(:,i) = vi ./ nrm;
end
```

2. Generating Co-variance Matrix

```
Sigma = U * diag((1:n).^(-alpha)) * U';
```

3. Generating 10 n-dimensional vectors

```
xs = mvnrnd(zeros(n,1), Sigma, 10)';
```

Generating measured signal

1. Generating a random sensing matrix

```
phi = randn(m, n) / sqrt(m);
```

2. Generating noisy measurement

```
y = phi * xs(:, i);
sig = 0.01 * mean(abs(y));
y = y + mvnrnd(zeros(m, 1), eye(m)*sig^2, 1)';
```

Reconstruction

1. Variables used in reconstruction (for faster computation)

```
SpT = Sigma * phi';
pSpT = phi * SpT;
```

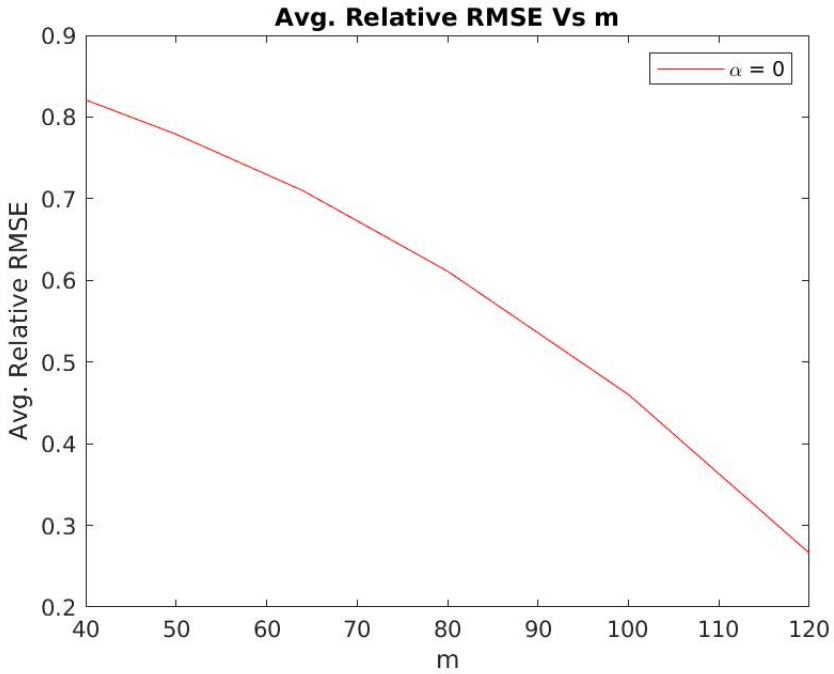
2. Reconstruction of x

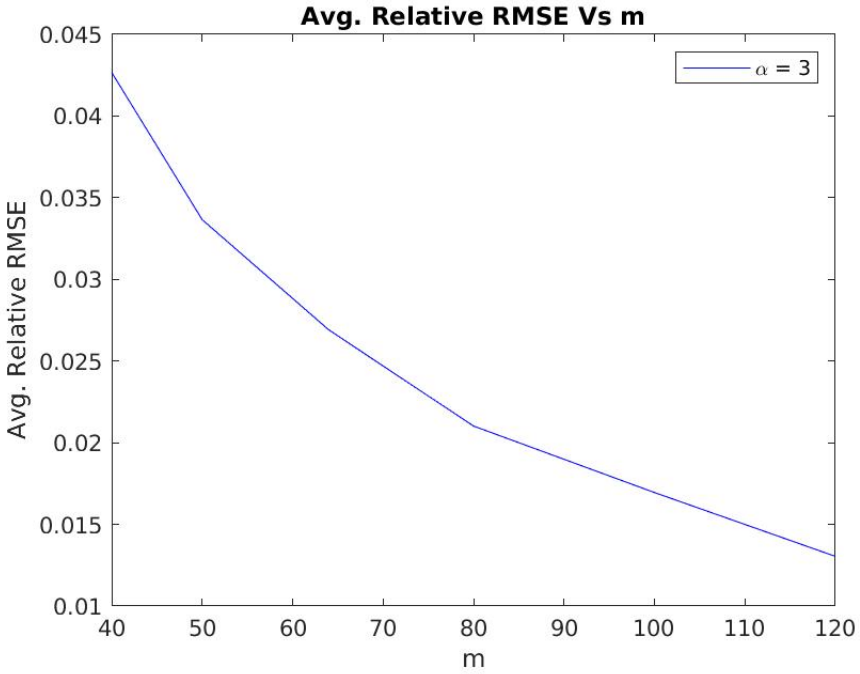
```
x = (SpT - SpT/(eye(m)*sig^2 + pSpT)*pSpT) * y / (sig^2);
```

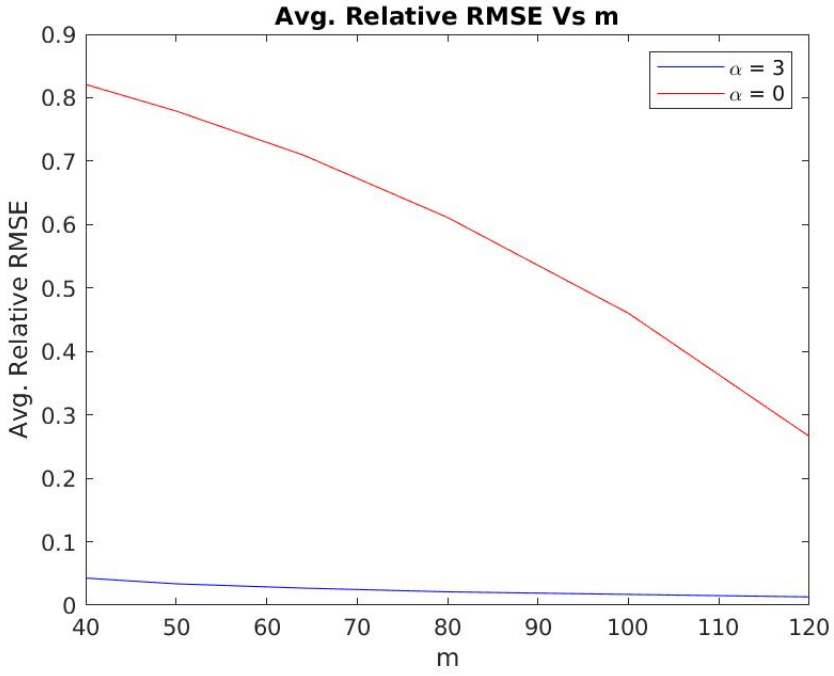
3. Calculating Avg. Relative RMSE

```
rmse(i) = sqrt(mean((x - xs(:,i)).^2))/sqrt(mean(xs(:,i).^2));
```

Results




Avg. Relative RMSE Vs m
Avg. Relative RMSE
m
α = 3
0.045
0.04
0.035
0.03
0.025
0.02
0.015
0.01
40
50
60
70
80
90
100
110
120


Avg. Relative RMSE Vs m
Avg. Relative RMSE
m
α = 3
α = 0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
40
50
60
70
80
90
100
110
120

$m(< n = 128)$	Avg. Relative RMSE $\left(\frac{1}{10}\sum_{i=1}^{10}\frac{\|\hat{\boldsymbol{x}}_{\boldsymbol{i}} - \boldsymbol{x}_{\boldsymbol{i}}\|_2}{\|\boldsymbol{x}_{\boldsymbol{i}}\|_2}\right)$	
	$\alpha = 0$	$\alpha = 3$
40	0.820906	0.042668
50	0.778914	0.033657
64	0.709862	0.026905
80	0.611197	0.021020
100	0.460178	0.016950
120	0.265929	0.013042

Comments

- Average Relative RMSE decreases with increase in m as expected. We have more information about the original signal, so reconstruction is better.
- Average Relative RMSE for $\alpha = 3$ lies in range 0.01-0.05. As the eigenvalues decay very fast (cubic speed), so several elements are zero (or almost zero) with high probability. This means that the signal $\boldsymbol{x}$ is sparse, thus giving better reconstruction.
- Average Relative RMSE for $\alpha = 0$ lies in range 0.25-0.85. As the eigenvalues doesn't decay, so all elements are non-zero with high probability. This means that the signal $\boldsymbol{x}$ is not sparse, thus giving poorer reconstruction.
- Average Relative RMSE for any m is higher for smaller α (decay factor)

Question 3

Q1.

Since $X^\star$ is the optimal solution of $\arg\min_X \|X\|_\star$ s.t. $\mathcal{A}(X) = b$ (Eq 3.1) and $\mathcal{A}(X_0) = b$,

$$\|X^\star\|_\star \leq \|X_0\|_\star$$

Q2.

By construction, X_0 and R_c have same dimension.
From Lemma 3.4, we have $X_0 R_c' = 0$ and $X_0' R_c = 0$.
Thus, X_0 and R_c satisfy the hypothesis for Lemma 2.3.
So, we have

$$\|X_0 + R_c\|_\star = \|X_0\|_\star + \|R_c\|_\star \implies \|X_0 + R_c\|_\star - \|R_0\|_\star = \|X_0\|_\star + \|R_c\|_\star - \|R_0\|_\star$$

Q3.

As we have obtained σ by singular value decomposition of R_c,
σ_i's are the singular values of R_c with $\sigma_i \geq \sigma_j \ \forall\ 1 \leq i \leq j \leq rank(R_c)$.

$$\sigma_j \geq \sigma_k \quad \forall j \in I_i = \{3r(i-1)+1, \dots, 3ri\},\ \forall k \in I_{i+1} = \{3ri+1, \dots, 3r(i+1)\}$$

$$\sum_{j \in I_i} \sigma_j \geq 3r\sigma_k \implies \sigma_k \leq \frac{1}{3r} \sum_{j \in I_i} \sigma_j \quad \forall k \in I_{i+1}$$

Q4.

$$\|R_{i+1}\|_F^2 = \sum_{j \in I_{i+1}} \sigma_j^2 \quad \text{(From definition of Frobenius norm on Page 6 of the paper)}$$

$$\|R_i\|_\star = \sum_{j \in I_i} \sigma_j \quad \text{(From definition of Nuclear norm on Page 6 of the paper)}$$

$$\sigma_k \leq \frac{1}{3r} \sum_{j \in I_i} \sigma_j \ \forall\ k \in I_{i+1} \quad \text{(From Eq 3.5, also Q3)}$$

$$\|R_{i+1}\|_F^2 = \sum_{j \in I_{i+1}} \sigma_j^2 \leq \sum_{j \in I_{i+1}} \frac{1}{9r^2} \left(\sum_{k \in I_i} \sigma_k \right)^2 = \frac{1}{3r} \left(\sum_{k \in I_i} \sigma_k \right)^2 = \frac{1}{3r} \|R_i\|_\star^2$$

Q5.

$$\|R_{i+1}\|_F = \frac{1}{\sqrt{3r}} \|R_i\|_\star \quad \text{(Starting from Q4)}$$

We add the equations for $i \in \{1, 2, \dots\}$ to get

$$\sum_{j_{max} \geq j \geq 2} \|R_j\|_F \leq \sum_{j_{max}-1 \geq j \geq 1} \frac{1}{\sqrt{3r}} \|R_j\|_\star \leq \sum_{j_{max} \geq j \geq 1} \frac{1}{\sqrt{3r}} \|R_j\|_\star$$

(Singular values are non-negative, so is their sum - Nuclear norm)

Q6.

$$\|R_0\|_\star \geq \|R_c\|_\star \quad \text{(From Eq 3.4)}$$
$$\frac{1}{\sqrt{3r}}\|R_c\|_\star \leq \frac{1}{\sqrt{3r}}\|R_0\|_\star$$

Q7.

$$\begin{aligned}\|R_0\|_\star &\leq \sqrt{rank(R_0)}\|R_0\|_F \quad \text{(From Eq 2.1 on Page 6 of the paper)}\\ &\leq \sqrt{2rank(X_0)}\|R_0\|_F = \sqrt{2r}\|R_0\|_F \quad \text{(Lemma 3.4)}\end{aligned}$$

$$\frac{1}{\sqrt{3r}}\|R_0\|_\star \leq \frac{\sqrt{2r}}{\sqrt{3r}}\|R_0\|_F$$

Q8.

As mentioned on Page 9 of the paper, "For matrices, the rank function is subadditive."
$\|R_0 + R_1\|_\star \leq \|R_0\|_\star + \|R_1\|_\star$.
From Lemma 3.4, $\|R_0\|_\star \leq 2r$.
From construction of R_1, $\|R_1\|_\star \leq 3r$.
So, $\|R_0 + R_1\|_\star \leq 5r$, i.e., rank of $R_0 + R_1$ is at most $5r$.

Q9.

$$\begin{aligned}\|\mathcal{A}(R)\| &= \|\mathcal{A}(R_0 + R_1 + \sum_{j\geq 2} R_j)\| \quad \text{(By construction of } R_0 \text{ and } R_c)\\ &= \|\mathcal{A}\left(R_0 + R_1\right) + \sum_{j\geq 2}\mathcal{A}(R_j)\| \quad (\mathcal{A} \text{ is a linear map})\\ &\geq \|\mathcal{A}\left(R_0 + R_1\right)\| - \|\sum_{j\geq 2}\mathcal{A}(R_j)\| \quad \text{(Triangle inequality)}\\ &\geq \|\mathcal{A}\left(R_0 + R_1\right)\| - \sum_{j\geq 2}\|\mathcal{A}(R_j)\| \quad \text{(Triangle inequality)}\end{aligned}$$

Q10.

Using the definition of restricted isometry constant (Eq 3.2),

$(1 - \delta_{5r})\|R_0 + R_1\|_F \leq \|\mathcal{A}(R_0 + R_1)\| \leq (1 + \delta_{5r})\|R_0 + R_1\|_F$ (From Q8, $R_0 + R_1$ has rank at most $5r$)
$(1 - \delta_{3r})\|R_i\|_F \leq \|\mathcal{A}(R_i)\| \leq (1 + \delta_{3r})\|R_i\|_F \; \forall i \geq 2$ (By construction, R_is have rank at most $3r$)

$$\begin{aligned}\|\mathcal{A}(R)\| &\geq \|\mathcal{A}\left(R_0 + R_1\right)\| - \sum_{j\geq 2}\|\mathcal{A}(R_j)\| \quad \text{(Starting from Q9)}\\ &\geq (1 - \delta_{5r})\|R_0 + R_1\|_F - \sum_{j\geq 2}(1 + \delta_{3r})\|R_i\|_F\\ &\geq (1 - \delta_{5r})\|R_0 + R_1\|_F - (1 + \delta_{3r})\sum_{j\geq 2}\|R_i\|_F\end{aligned}$$

Q11.

The assumption is that $X^\star$ is the optimal solution of $\arg\min_X \|X\|_\star$ s.t. $\mathcal{A}(X) = b$ where, $b := \mathcal{A}(X_0)$.

$$\begin{aligned}\mathcal{A}(R) &= \mathcal{A}(X^\star - X_0) \quad \text{(By definition of R)} \\ &= \mathcal{A}(X^\star) - \mathcal{A}(X_0) \quad (\mathcal{A} \text{ is a linear map}) \\ &= b - b = 0\end{aligned}$$

Q12.

$$\left((1-\delta_{5r}) - \frac{9}{11}(1+\delta_{3r})\right)\|R_0\|_F > 0$$

$$\left((1-\delta_{5r}) - \frac{9}{11}(1+\delta_{3r})\right) > 0 \quad \text{(Frobenius norm is non-negative)}$$

$$(11 - 11\delta_{5r} - 9 - 9\delta_{3r}) > 0$$

$$(2 - 11\delta_{5r} - 9\delta_{3r}) > 0$$

$$2 > 11\delta_{5r} + 9\delta_{3r}$$

Question 4

Part (1)

Question:

Why do the theorems on low rank matrix completion require that the singular vectors be incoherent with the canonical basis (i.e. columns of the identity matrix)?

Answer:

Need

To recover a low-rank matrix, this matrix cannot be in the null space of the sampling operator giving the values of a subset of the entries.

It is easy to see that if the singular vectors of a matrix $\boldsymbol{M}$ are highly concentrated, then $\boldsymbol{M}$ could very well be in the null-space of the sampling operator.

For example, consider a matrix with all entries zero, except the top right entry, which is set to 1.
One would basically need to see all the entries of M to be able to recover this matrix exactly by any method whatsoever. There is an endless list of examples of this sort.

Hence, we arrive at the notion that, somehow, the singular vectors need to be sufficiently *spread out*.

This motivates the following definition.

Coherence

The coherence of subspace $\boldsymbol{U}$ of $\mathbb{R}^n$ of dimension r ($\boldsymbol{U}$ has size $n \times r$) with respect to the canonical basis e_i is defined as follows:

$$\mu(\boldsymbol{U}) = \frac{n}{r} \max_{1 \leq i \leq n} \|P_{\boldsymbol{U}} e_i\|^2$$

Here $P_{\boldsymbol{U}}$ is the orthogonal projection onto $\boldsymbol{U}$, that is $P_{\boldsymbol{U}} = \boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^T$

The Condition

We need the singular vectors to be uncorrelated with the standard basis—in order to minimize the number of observations needed to recover a low-rank matrix.

Both the left and right singular vectors need to be uncorrelated with the standard basis.

Matrices whose column and row spaces have low coherence cannot really be in the null space of the sampling operator.

Suppose the sampling matrix $\boldsymbol{M}$ has its singular value decomposition as $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$
Then, we require a bound on the coherence of both matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ to be bounded by some constant.

That is, there should exist a positive number μ_0 such that

$$\max\{\mu(\boldsymbol{U}), \mu(\boldsymbol{V})\} \leq \mu_0$$

Part (2)

Question:

How would this coherence condition change if the sampling operator were changed to the one in the equation 1.13 of the paper?

Answer:

The Sampling Operator

We have two orthonormal bases $\boldsymbol{f_1, \ldots, f_n}$ and $\boldsymbol{g_1, \ldots, g_n}$ of $\mathbb{R}^n$, and we are interested in solving the following: rank minimization problem

$$\begin{aligned} \text{minimize} \quad & rank(\boldsymbol{X}) \\ \text{subject to} \quad & \boldsymbol{f_i^* X g_j} = \boldsymbol{f_i^* M g_j} \quad \forall\, (i,j) \in \Omega \end{aligned}$$

The Condition

Now that we have re-defined our sampling operator, we now no longer need our sampled matrix to be incoherent with the canonical basis.

Now, we need our rows and columns to have incoherence with respect to these new bases.

We can express this condition as follows:

Firstly, we note that there exist orthonormal(unitary for complex matrices) transformations $\boldsymbol{F}$ and $\boldsymbol{G}$ such that for all $i = 1, 2, \ldots n$, we get $\boldsymbol{e}_i = \boldsymbol{F f}_i$ and also $\boldsymbol{e}_i = \boldsymbol{G g}_i$.

Then, we can write:

$$\boldsymbol{f_i^* X g_j} = \boldsymbol{e_i^* (F X G^*) e_j}$$

Hence, we need the conditions of theorems to hold for this matrix $\boldsymbol{FXG^*}$ now.

In other words, all that is needed is that the column and row spaces of $\boldsymbol{M}$ be respectively incoherent with the basis $(\boldsymbol{f}_i)$ and $(\boldsymbol{g}_i)$.

Question 5

Answer:

Details of the Paper

Title of the Paper: Finding correspondence from multiple images via sparse and low-rank decomposition
A link to the paper: `https://link.springer.com/content/pdf/10.1007/978-3-642-33715-4_24.pdf`
Authors: Zinan Zeng, Tsung-Han Chan, Kui Jia, Dong Xu
Publisher: Springer, Berlin, Heidelberg
Conference: European Conference on Computer Vision
Publication date: 2012/10/7

The problem being solved in the paper

The correspondence problem refers to the task of finding a set of points in one image which can be identified as the same points in another image. To do this, points or features in one image are matched with the corresponding points or features in another image.

The problem of finding the correspondence from multiple images, is a challenging combinatorial problem, and it is the one being discussed in this paper.

Finding visual pattern correspondence across images or video sequences is a long-standing problem in computer vision. It facilitates a wide range of vision applications such as object recognition, 3D reconstruction and image matching.

Here, we focus on the problem to find the feature correspondence across video sequence.

Finding the global correspondence for features/patches (referred to as pat- terns) from multiple images can be formulated as a combinatorial NP-hard problem, in which a set of partial permutation matrices need to be found. Meanwhile, if each pattern is exactly the same across images, the matrix formed by the well corresponded patterns will be low rank.

For low rank representation we can incorporate the sparse error term into the frameworks to cope with corruption and occlusion that inevitably exist in images and videos.

Problem Statement

Partial Permutation Matrices: The set $\mathcal{P}_n$ of partial permutation matrices can be defined as follows:

$$\mathcal{P}_n = \{\boldsymbol{P}_n \quad | \, \boldsymbol{P}_n \in \{0,1\}^{K_n \times K}, \quad \mathbf{1}_{K_n}^T \boldsymbol{P_n} = \mathbf{1}_K^T, \quad \boldsymbol{P}_n \mathbf{1}_K \leq \mathbf{1}_{K_n}\}$$

where: $\{0,1\}^{K_n \times K}$ denotes a $K_n \times K$ matrix whose elements are either 0 or 1 and $\mathbf{1}_c$ denotes a column vector of all 1 of length c.

Suppose we are given N images. For each image I_n we extract K_n features of size d,
$\boldsymbol{f}_{n,1}, \ldots, \boldsymbol{f}_{n,K} \in \mathbb{R}^d$ at K_n landmark locations, and stack them as a matrix $\boldsymbol{F_n} = [\boldsymbol{f}_{n,1} \cdots \boldsymbol{f}_{n,K_n}] \in \mathbb{R}^{d \times K_n}$
Our interest is to find K intrinsic patterns for each image such that for all n, $K \leq K_n$, and their correspondence among N images (i.e., N sets of K features).
There exist partial permutation matrices $\boldsymbol{P}_i$ such that the reordered patterns are well corresponded.

That is, The matrix $A = [vec(\boldsymbol{F_1 P_1}) || vec(\boldsymbol{F_N P_N})] \in \mathbb{R}^{dK \times N}$ should be approximately low-rank.

Moreover, in practice, error such as corruption and occlusion is common in images and videos, thus the low rank property of the aligned matrix is likely to be violated. To improve the robustness, such error is modeled as a sparse matrix as it only affects a small fraction of the data.