ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

SUPER-RESOLUTION OFF THE GRID

EPFL - SEMESTER PROJECT IN COMPUTER SCIENCE

Sébastien Ollquist

1st April 2022

ABSTRACT

Super-Resolution is the tool that allows us to upgrade the quality of images. The goal of this project is to study a recent paper on super-resolution and discuss the main algorithm in order to eventually find an improvement using information theoretical bounds.

> *In theory, theory and practice are the same. In practice, they are not.*
>
> Albert Einstein

CHAPTER 1

INTRODUCTION

With the increasing amount of data nowadays, it has become extremely important to develop tools that allow us to treat it better. In the case of images, unfortunately a lot of them are still taken with old technology leading to a weak resolution. Recently, a new theory called super-resolution has emerged. Its aim is to reconstruct an image of better quality from a poor quality image.

The paper we will study here is called "Super-Resolution Off the Grid" and has been published by Qingqing Huang and Sham M. Kakade in September 2015. From a theoretical point of view, super-resolution is the problem of recovering a superposition of point sources using bandlimited measurements, which may be corrupted with noise.

The goal of this project is to get familiar with the theory of super-resolution along with all important mathematical preliminaries and study the main result presented in the above mentioned paper. The idea is then to compare this work with other related ones and discuss the efficiency of the main algorithm. Ultimately, it would be a nice addition to start working on an improvement of the main algorithm in order to reduce its complexity and thus make it run faster.

CHAPTER 2

MATHEMATICAL REFRESHER

2.1 A MATHEMATICAL THEORY OF SUPER-RESOLUTION

We consider k point sources in d dimensions, where the points are separated by a distance at least Δ (in Euclidean distance). The d-dimensional signal $x(t)$ can be modeled as a weighted sum of k Dirac measures in $\mathbb{R}^d$ as

$$x(t) = \sum_{j=1}^{k} w_j \delta_{\mu^{(j)}},$$

where the $\mu^{(j)}$'s are the point sources in $\mathbb{R}^d$ and $w_j \in \mathbb{C}$ the weights such that $|w_j| < C$ for every $j \in [k]$ and some absolute constant $C > 0$.

2.2 MAIN TOOLS

The main mathematical tools that are needed to understand the paper essentially lie around the subject of linear algebra. They include operations related to vectors, matrices and tensors. We also require some probabilistic analysis tools since the algorithm is partly random. In this chapter, we introduce those tools, prove the more important results and closely relate them to the paper.

2.2.1 GENERALISED EIGENVALUE PROBLEM

Before introducing the generalised eigenvalue problem, it is important to recall what a condition number is for a particular matrix. Suppose for instance we have a matrix $X \in \mathbb{R}^{m \times n}$. We let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of $X^T X$ (with repetitions) and arrange them so that $\lambda_1 \geq \ldots \geq \lambda_n \geq 0$. Then, the $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$ such that $\sigma_i = \sqrt{\lambda_i}$ are called the *singular values* of X. Define $\sigma_{max}(X) = \sigma_1$ and $\sigma_{min}(X) = \sigma_n$. We then define the condition number of a matrix to be the ratio between the largest and the smallest singular value of X. That is

$$\mathrm{cond}_2(X) = \sigma_1/\sigma_n = \frac{\sigma_{max}(X)}{\sigma_{min}(X)}. \tag{2.1}$$

The above factor governs the noise tolerance of the generalised eigenvalue problem, i.e. it is the measure of sensitiveness of a matrix to arbitrary perturbations. Taking its limiting value will allow us to state whether or not the main algorithm achieves stable recovery of the point sources.

The goal of an eigenvalue problem is to simply find the eigenvalues of a particular matrix A. To do so, we generally solve the following equation: $AU = UV \Rightarrow AUU^T = UVU^T \Rightarrow A = UVU^T = UVU^{-1}$,

where $\mathbb{R}^{d\times d} \ni U := (u_1, \ldots, u_d)$ are the eigenvectors and $\mathbb{R}^{d\times d} \ni V := Diag\left[(\lambda_1, \ldots, \lambda_d)^T\right]$ the eigenvalues. However, in the generalised version of this problem, we add another random matrix B such that the problem becomes

$$AU = BUV \Rightarrow AUU^T = BUVU^T \Rightarrow A = BUVU^T = BUVU^{-1},$$

where U again are the eigenvectors and V the eigenvalues. Note that A and B are both symmetric. We form the pair (A, B) called the *pencil* and the pair (U, V) called the *eigenpair*. Therefore, we introduce a first version of what is called the matrix pencil method, in which given a pair of matrices (A, B), we wish to find the generalised eigenvalues λ for which there is a vector $\mathbf{x}$ such that $A\mathbf{x} = \lambda B\mathbf{x}$. Note that the eigenvalues of A are the solution to the generalised eigenvalue problem where $B = I$.

2.3 TENSOR DECOMPOSITION

2.3.1 GENTLE INTRODUCTION

A tensor is a generalisation of a matrix to more than two dimensions. We can think of a tensor as a point in $\mathbb{C}^{m_1\times\ldots\times m_k}$ where k is the order of the tensor. Most of the time here, $k = 3$ since three dimensions suffice for our analysis. Note that if T is an order three tensor of dimensions $m_A \times m_B \times m_C$, we can view it as a collection of m_C matrices of size $m_A \times m_B$ stacked on top of each other. For example, the entry $A_{i,j,k}$ of a 3-tensor A will simply be the (i, j)'th entry of the k'th matrix.

We define the *rank* of a tensor V as the minimum r such that we can write V as the sum of rank one tensors. A rank one tensor will be decomposed in the form of a tensor product of three matrices A, B and C as $V = A \otimes B \otimes C$. The above product is element-wise defined as $V_{i_1,i_2,i_3} = \sum_{j=1}^{k} A_{i_1,j}B_{i_2,j}C_{i_3,j}$.

An alternative definition is given using the notion of a multi-linear mapping. Namely, for given dimensions m_A, m_B, m_C, the mapping $V(\cdot,\cdot,\cdot) : \mathbb{C}^{m\times m_A} \times \mathbb{C}^{m\times m_B} \times \mathbb{C}^{m\times m_C} \to \mathbb{C}^{m_A\times m_B\times m_C}$ is defined as:

$$[V(X_A, X_B, X_C)]_{i_1,i_2,i_3} = \sum_{j_1,j_2,j_3\in[m]} V_{j_1,j_2,j_3}[X_A]_{j_1,i_1}[X_B]_{j_2,i_2}[X_C]_{j_3,i_3}.$$

We can verify that for a particular vector $a \in \mathbb{C}^m$, the projection $V(I, I, a)$ of V along the 3rd dimension is $V(I, I, a) = ADiag(C^T a)B^T$ as long as V admits a tensor decomposition $V = A \otimes B \otimes C$. Indeed,

$$\begin{aligned}
[V(I,I,a)]_{i_1,i_2} &= \sum_{j_1,j_2,j_3\in[m]} V_{j_1,j_2,j_3}[I]_{j_1,i_1}[I]_{j_2,i_2}[a]_{j_3} \\
&= \sum_{j_1,j_2,j_3\in[m]} \sum_{n\in[k]} A_{j_1,n}B_{j_2,n}C_{j_3,n}[I]_{j_1,i_1}[I]_{j_2,i_2}[a]_{j_3} \\
&= \sum_{j_3\in[m]} \sum_{n\in[k]} A_{i_1,n}B_{i_2,n}C_{j_3,n}[a]_{j_3} \\
&= \sum_{n\in[k]} A_{i_1,n}B_{i_2,n} \sum_{j_3\in[m]} C_{j_3,n}[a]_{j_3} \\
&= \left[ADiag(C^T a)B^T\right]_{i_1,i_2}.
\end{aligned}$$

Since the above is true for every i_1, i_2, we get the desired equality.

2.3.2 JENNRICH'S ALGORITHM

As previously stated, the goal of tensor decomposition is to, given a tensor T of rank k, decompose it as a sum of rank 1 tensors of appropriate dimensions. Jennrich's algorithm is commonly used for tensor decomposition. We denote the following theorem:

Theorem 1 *Let $T = \sum_{j=1}^{k} u_i \otimes v_i \otimes w_i$ be a tensor in which each set of vectors $\{u_i\}_i$ and $\{v_i\}_i$ are linearly independent. Moreover, each pair of vectors in $\{w_i\}_i$ are also linearly independent. Then, the above decomposition is unique up to rescaling, and there is an efficient algorithm to find it.*

The aforementioned algorithm is described in Algorithm 1. Observe that

Algorithm 1 Jennrich's algorithm for tensor decomposition

Input: a tensor $\tilde{F} \in \mathbb{C}^{m \times m \times 3}$ of rank k.
Choose random unit vectors $a, b \in \mathbb{R}^m$.
Compute $\tilde{F}(I, I, a) = UD_aV^T$, where $D_a = Diag(\langle w^{(i)}, a \rangle)$.
Compute $\tilde{F}(I, I, b) = UD_bV^T$, where $D_b = Diag(\langle w^{(i)}, b \rangle)$.
Compute the diagonalisations $\tilde{F}(I, I, a)\tilde{F}(I, I, b)^{-1}$ and $\tilde{F}(I, I, b)\tilde{F}(I, I, a)^{-1}$.
Solve the linear system to recover the w_j's.
Return U, V, W.

$$\tilde{F}(I, I, a)\tilde{F}(I, I, b)^{-1} = UD_aV^T(V^T)^{-1}D_b^{-1}U^{-1} = UD_aD_b^{-1}U^{-1},$$

and similarly

$$\tilde{F}(I, I, a)^{-1}\tilde{F}(I, I, b) = (V^T)^{-1}D_a^{-1}U^{-1}UD_bV^T = (V^T)^{-1}D_a^{-1}D_bV^T.$$

The correctness of the algorithm follows from the uniqueness of eigendecomposition of a matrix when the eigenvalues are distinct. For a random choice of a and b (in our case we choose the basis vectors e_1 and e_2), with high probability the eigendecompositions are unique so we can recover the u_i's and the v_i's easily by simply recovering the columns of U and V, respectively.

CHAPTER 3

INITIAL WORK

3.1 PRONY'S METHOD

3.2 UNIVARIATE CASE: ADAPTATION OF PRONY'S METHOD

The initial setting is as follows. We construct two $m \times m$ complex valued Hankel matrices H_0 and H_1, that is, matrices such that their skew-diagonals[1] are constants. We have $D_w \in \mathbb{C}^{k\times k}$ where $[D_w]_{j,j} = w_j$ and $D_\mu \in \mathbb{C}^{k\times k}$ where $[D_\mu]_{j,j} = e^{i\pi\mu^{(j)}}$. We furthermore have a Vandermonde matrix $V_m \in \mathbb{C}^{m\times k}$ defined as

$$V_m = \begin{pmatrix} 1 & \dots & 1 \\ \left(e^{i\pi\mu^{(1)}}\right)^1 & \dots & \left(e^{i\pi\mu^{(k)}}\right)^1 \\ \vdots & \dots & \vdots \\ \left(e^{i\pi\mu^{(1)}}\right)^{m-1} & \dots & \left(e^{i\pi\mu^{(k)}}\right)^{m-1} \end{pmatrix}$$

We construct a 3rd order tensor $F \in \mathbb{C}^{m\times m\times 2}$ in which $F_{i,i',j} = [H_{j-1}]_{i,i'}$ for $j = 1, 2$ and $i, i' \in [m]$.

Fact 2 *In the setting above, F admits the unique rank k tensor decomposition $F = V_m \otimes V_m \otimes (V_2 D_w)$.*

Proof We start by computing the product $V_2 D_w$. By definition of V_m, we have that $[V_2]_{r,c} = \left(e^{i\pi\mu^{(c)}}\right)^{r-1}$ for $r \in [2]$ and $c \in [k]$. Multiplying with the diagonal matrix D_w yields

$$[V_2 D_w]_{r,c} = w_c\left(e^{i\pi\mu^{(c)}}\right)^{r-1}, \quad \forall r \in [2], c \in [k].$$

We aim to show that

$$\begin{aligned} F_{i_1,i_2,i_3} &= \sum_{n=1}^{k} [V_m]_{i_1,n}[V_m]_{i_2,n}[V_2 D_w]_{i_3,n} \\ &= \sum_{n=1}^{k} \left(e^{i\pi\mu^{(n)}}\right)^{i_1-1}\left(e^{i\pi\mu^{(n)}}\right)^{i_2-1} w_n \left(e^{i\pi\mu^{(n)}}\right)^{i_3-1} \\ &= \sum_{n=1}^{k} w_n \left(e^{i\pi\mu^{(n)}}\right)^{i_1+i_2+i_3-3}, \end{aligned}$$

[1] A skew-diagonal is the diagonal in the North-East direction.

for $i_1, i_2 \in [m]$ and $i_3 \in [2]$. It can be verified that the measurements that form both Hankel matrices are given by

$$f(s) = \sum_{j \in [k]} w_j \big(e^{i\pi\mu^{(j)}}\big)^s,$$

with $s = 0, \ldots, 2m-1$ for H_0 and $s = 1, \ldots, 2m$ for H_1. For the details of the proof, see Appendix A. Then, since $F_{i_1,i_2,i_3} = [H_{i_3-1}]_{i_1,i_2}$, for a fixed i_3 we have either $F_{i_1,i_2,1} = [H_0]_{i_1,i_2}$ or $F_{i_1,i_2,2} = [H_1]_{i_1,i_2}$. Note that since we are dealing with Hankel matrices, $[H_0]_{i_1,i_2} = f(i_1) + f(i_2) - 2 = f(i_1 + i_2 - 2)$. Hence,

$$F_{i_1,i_2,i_3} = \sum_{n \in [k]} w_n \big(e^{i\pi\mu^{(n)}}\big)^{i_1+i_2+i_3-3},$$

as required. ■

3.3 MULTIVARIATE TOY CASE

We follow the definitions given in the paper. We have $D_w = Diag(w_j) \in \mathbb{C}^{k\times k}$ and $[V_d]_{r,c} = e^{i\pi\mu_r^{(c)}} \in \mathbb{C}^{d\times k}$ the latter in which $\mu_n^{(i)}$ are i.i.d. $\sim Unif([-1,+1])$. Furthermore, we have $F_{n_1,n_2,n_3} = f(s)\big|_{s=e_{n_1}+e_{n_2}+e_{n_3}}$, for all $n_1, n_2, n_3 \in [d]$. We have the following fact:

Fact 3 *In the setting above, F admits the tensor decomposition $F = V_d \otimes V_d \otimes (V_d D_w)$.*

Proof We wish to show that $f(e_1+e_2+e_3) = \sum_{j=1}^k w_j e^{i\pi(\mu_1^{(j)}+\mu_2^{(j)}+\mu_3^{(j)})}$. To do so, we first compute the matrix product $V_d D_w$. Since $[V_d]_{r,c} = e^{i\pi\mu_r^{(c)}}$ for $r \in [d], c \in [k]$ and $D_w = Diag(w_j)$, we directly have that

$$[V_d D_w]_{r,c} = w_c e^{i\pi\mu_r^{(c)}}, \quad r \in [d], c \in [k],$$

so that by definition of tensor decomposition, we get

$$\begin{aligned} F_{n_1,n_2,n_3} &= \sum_{j=1}^k [V_d]_{n_1,j}[V_d]_{n_2,j}[V_d D_w]_{n_3,j} \\ &= \sum_{j=1}^k e^{i\pi\mu_{n_1}^{(j)}} e^{i\pi\mu_{n_2}^{(j)}} w_j e^{i\pi\mu_{n_3}^{(j)}} \\ &= \sum_{j=1}^k w_j e^{i\pi(\mu_{n_1}^{(j)}+\mu_{n_2}^{(j)}+\mu_{n_3}^{(j)})} = f(e_1+e_2+e_3), \end{aligned}$$

as required. ■

CHAPTER 4

MAIN ALGORITHM

Let us recall what the goal of the problem is. Broadly speaking, super-resolution aims to recover a superposition of point sources using bandlimited measurements that may be corrupted with noise. In this setting, the created algorithm works in the Fourier domain where each of the k points of the d-dimensional plane are separated by a distance at least Δ. Hence, the frequencies of the Fourier measurements are bounded by $O(1/\Delta)$. The idea of the procedure is to take random bandlimited measurements with cutoff frequency bounded by $\Omega(\sqrt{d}/\Delta)$ and perform Tensor decomposition to recover the estimates of the point sources.

4.1 ALGORITHM DESCRIPTION

INPUT AND OUTPUT The algorithm takes as input a cutoff frequency R, the defined number of measurements m and a noisy measurement function $\tilde{f}(\cdot)$ and outputs the set of estimates

$$\{\widehat{w}_j, \widehat{\mu}^{(j)} : j \in [k]\},$$

where the $\widehat{w}_j$'s are the complex weight coefficients and the $\widehat{\mu}^{(j)}$'s are the estimates of the point sources. Note that in case the noise is non-existant (i.e. when $\epsilon_z = 0$), the parameters are recovered exactly. Otherwise, stable recovery implies that the estimates are such that

$$\min_{\pi} \max \left\{ ||\widehat{\mu}^{(j)} - \mu^{(\pi(j))}||_2 : j \in [k] \right\} \leq poly(d, k)\epsilon_z.$$

In words, the estimates of the point sources differ from their real values by at most the noise ϵ_z scaled by a polynomial function that depends on the number of dimensions d and the number of point sources k.

MEASUREMENTS We first generate a set $\mathcal{S} = \{s^{(1)}, \ldots, s^{(m+n+1)}\}$ where $s^{(1)}, \ldots, s^{(m)}$ are m i.i.d. samples from the Gaussian distribution $\mathcal{N}(0, R^2 I_{d\times d})$, $s^{(m+n)} = e_n$ for all $n \in [d]$ and $s^{(m+n+1)} = 0$ and we let $m' = m + d + 1$. We then take a sample v from the unit sphere and set $v^{(1)} = v$ and $v^{(2)} = 2v$. Since we are in dimension d, the sample v is such that $x_1^2 + \ldots + x_d^2 = 1$. Finally, we construct a tensor $\tilde{F} \in \mathbb{C}^{m'\times m'\times 3} : \tilde{F}_{n_1,n_2,n_3} = \tilde{f}(s)|_{s=s^{(n_1)}+s^{(n_2)}+v^{(n_3)}}$.

TENSOR DECOMPOSITION We apply Jennrich's algorithm on $\tilde{F}$ to obtain the estimates $\widehat{V}_{S'}$ and $\widehat{D}_w$. We then normalise each value of $\widehat{V}_{S'}$ so that the last element of each column is 1.

READ OF ESTIMATES We finish by recovering the real part of the estimates of the point sources by setting $\widehat{\mu}^{(j)} = Real(\log([\widehat{V}_S]_{[m+1:m+d,j]})/(i\pi))$ for every $j \in [k]$ and we find the best possible matching coefficients by setting $\widehat{W} = \arg\min_{W \in \mathbb{C}^k} ||\widehat{F} - \widehat{V}_{S'} \otimes \widehat{V}_{S'} \otimes \widehat{V}_d D_w||_F$, where $|| \cdot ||_F$ is the Frobenius norm of $(\cdot)$.

4.2 ANALYSIS

First of all, note that the sample complexity is determined solely by the condition numbers of F and V_S.

CHAPTER 5

DISCUSSION

CHAPTER 6

IMPROVEMENT

CHAPTER 7

CONCLUSION

APPENDIX A

CONSTRUCTION OF HANKEL MATRICES

In this section, we prove that the Hankel matrices H_0 and H_1 defined earlier admit the respective diagonalisations $V_m D_w V_m^T$ and $V_m D_w D_\mu V_m^T$. To do so, we first compute the product $D_w V_m^T$. We have

$$
\begin{aligned}
D_w V_m^T &= \begin{pmatrix} w_1 & 0 & \dots & 0 \\ 0 & w_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_k \end{pmatrix} \begin{pmatrix} 1 & e^{i\pi\mu^{(1)}} & \dots & \left(e^{i\pi\mu^{(1)}}\right)^{m-1} \\ 1 & e^{i\pi\mu^{(2)}} & \dots & \left(e^{i\pi\mu^{(2)}}\right)^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{i\pi\mu^{(k)}} & \dots & \left(e^{i\pi\mu^{(k)}}\right)^{m-1} \end{pmatrix} \\
&= \begin{pmatrix} w_1 & w_1 e^{i\pi\mu^{(1)}} & \dots & w_1\left(e^{i\pi\mu^{(1)}}\right)^{m-1} \\ w_2 & w_2 e^{i\pi\mu^{(2)}} & \dots & w_2\left(e^{i\pi\mu^{(2)}}\right)^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_k & w_k e^{i\pi\mu^{(k)}} & \dots & w_k\left(e^{i\pi\mu^{(k)}}\right)^{m-1} \end{pmatrix},
\end{aligned}
$$

so that

$$
\begin{aligned}
H_0 = V_m D_w V_m^T &= \begin{pmatrix} 1 & \dots & 1 \\ e^{i\pi\mu^{(1)}} & \dots & e^{i\pi\mu^{(k)}} \\ \vdots & \dots & \vdots \\ \left(e^{i\pi\mu^{(1)}}\right)^{m-1} & \dots & \left(e^{i\pi\mu^{(k)}}\right)^{m-1} \end{pmatrix} \begin{pmatrix} w_1 & w_1 e^{i\pi\mu^{(1)}} & \dots & w_1\left(e^{i\pi\mu^{(1)}}\right)^{m-1} \\ w_2 & w_2 e^{i\pi\mu^{(2)}} & \dots & w_2\left(e^{i\pi\mu^{(2)}}\right)^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_k & w_k e^{i\pi\mu^{(k)}} & \dots & w_k\left(e^{i\pi\mu^{(k)}}\right)^{m-1} \end{pmatrix} \\
&= \begin{pmatrix} \sum_{j\in[k]} w_j & \sum_{j\in[k]} w_j e^{i\pi\mu^{(1)}} & \dots & \sum_{j\in[k]} w_j\left(e^{i\pi\mu^{(1)}}\right)^{m-1} \\ \sum_{j\in[k]} w_j e^{i\pi\mu^{(1)}} & \sum_{j\in[k]} w_j\left(e^{i\pi\mu^{(1)}}\right)^{2} & \dots & \sum_{j\in[k]} w_j\left(e^{i\pi\mu^{(1)}}\right)^{m} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j\in[k]} w_j\left(e^{i\pi\mu^{(1)}}\right)^{m-1} & \sum_{j\in[k]} w_j\left(e^{i\pi\mu^{(1)}}\right)^{m} & \dots & \sum_{j\in[k]} w_j\left(e^{i\pi\mu^{(1)}}\right)^{2m-1} \end{pmatrix} \\
&= \begin{pmatrix} f(0) & f(1) & \dots & f(m-1) \\ f(1) & f(2) & \dots & f(m) \\ \vdots & \vdots & \ddots & \vdots \\ f(m-1) & f(m) & \dots & f(2m-1) \end{pmatrix},
\end{aligned}
$$

where $f(s) = \sum_{j\in[k]} w_j \left(e^{i\pi\mu^{(j)}}\right)^s$ for $s = 0, \dots, 2m-1$ which indeed corresponds to the defined Hankel matrix. The proof is similar for H_1 except that s varies from 1 to $2m$ since element-wise multiplication of H_0 with the diagonal matrix D_μ yields a simple scaling by a factor $e^{i\pi\mu^{(j)}}$.