Abstract— Blind image deblurring (known as “blind” for not knowing, or partially knowing the deblurring filter) is a problem with great scientific interest, and with diverse applications (such as photography, biomedicine, surveillance, astronomy, etc.). It is a topic that has been discussed in many articles but, due to its complex nature, the solutions with promising results are still very scarce.

In this thesis, ADMM (Alternating direction method of multipliers) is implemented, in order to extend a blind image deblurring method, to simultaneously achieve the blind image deblurring with the super-resolution. Another objective is to obtain the possibility to make blind image deblurring in a scenario multi-frame. The final method allows combining multiple existing information in different blurred and low-resolution images, allowing estimation of a single image, which is both focused and of higher-resolution.

Index Terms— Blind Image Deblurring, Alternating Direction Method of Multipliers, multi-frame, super-resolution.

I. INTRODUCTION

A vast range of image deblurring is has applications, the most common being the blurred photographs, images that has degradation in photographic process with the use of digital cameras [1],[2] and [3]. The methods with image deblurring can be divided into two groups: Non-blind Image deblurring (NBID) and Blind Image Deblurring (BID). The Image deblurring - ID, consists in solving an inverse problem, whose goal is to recover an image that has undergone through a blurring degradation model with additive noise, assuming that the operator responsible for the linear degradation is known. Blind Image Deblurring [4] and [5], is the problem in which the operator of degradation is not known.

The main difficulty of blind image deblurring is related with that it is an ill-posed problem, since one only has access to the blurred image and there are an infinite number of pairs of solution “image + filter” that as a consequence of the inverse problem been misplaced. The process of degradation (blurring) on real images happens at the time of image acquisition, occurs a movement or image is out of focus that causes blurring. Blind Image deblurring can be combined with the Super-resolution problem, which consist in recovering one or more low resolution blurred images and estimate a high-resolution image. Figure 1 shows the scheme of Super-Resolution Blind Image deblurring. The direct problem consist in the degradation process, modeled by blurring the original image and then sampling the blurred image. The inverse problem, the problem we want to solve has one or several low-resolution blurred images and retrieve a high-resolution image without being blurred, and without having access to both the original image and the blurring filter used.

II. BLIND IMAGE DEBLURRING

The implemented method of BID is an extension based on method [4, 7], which the image and filter are estimated by using ADMM as an optimization tool.

A. Observation Model

The linear degradation of an image is modeled by:

\[ y = Ax + n, \]

where \( y \in \mathbb{R}^n \), \( x \in \mathbb{R}^m \) and \( n \in \mathbb{R}^n \) are column vectors representing, respectively, the degraded image, the original image and additive noise, all vectors ordered lexicographically. \( A \in \mathbb{R}^{m \times n} \) is a matrix responsible for the linear degradation of blurring a image with the blurring filter. In a cyclic convolution, we have \( n = m \) and \( A = H \in \mathbb{R}^{n \times n} \) is the square matrix corresponding to the linear cyclically convolution operator with a deblurring
filter, which can be lexicographically represented in vector $h$. However, in real situations the boundary conditions are unknown and to give a blurred image with $\sqrt{n} \times \sqrt{n}$ pixels, is necessary to have access to $(\sqrt{n} + 2l) \times (\sqrt{n} + 2l)$ pixels of the original image, and a deblurring filter with support $(2l + 1) \times (2l + 1)$ pixels. In this case, the deblurring operator $A = MH \in \mathbb{R}^{n \times (\sqrt{n} + 2l)^2}$ which can be separated in the product of a cyclic convolution performed by the matrix $H \in \mathbb{R}^{(\sqrt{n} + 2l)^2 \times (\sqrt{n} + 2l)^2}$ with matrix $M \in \mathbb{R}^{n^2 \times (\sqrt{n} + 2l)^2}$ excluding the edges of the pixels where the cyclical convolution is invalid:

$$y = Ax + n = MHx + n = MXh + n,$$  \hspace{1cm} (2)

where $h \in \mathbb{R}^m$ is the vector containing the deblurring filter ordered lexicographically and $X \in \mathbb{R}^{m \times m}$ is the square matrix corresponding to cyclic convolution with vector image $x \in \mathbb{R}^m$.

B. Cost Function

The BID method [4, 7] recovers the original image minimizing a cost function (2):

$$C(x, h) = \frac{1}{2} \|y - MHx\|_2^2 + \lambda \sum_i \left[\|D^{(i)}x\|_2^q + i_{S^+}(h)\right],$$  \hspace{1cm} (3)

where $i_{S^+}(h) = \begin{cases} 0, & h \in S^+, \\ \infty, & h \notin S^+ \end{cases}$, \hspace{1cm} (4)

is the indicator function that imposes filter $h$, to be in the set of vectors $S^+$, which have limited support and non negatives entries.

In the cost function $\lambda \sum_i \left[\|D^{(i)}x\|_2^q\]$ represents the regularization used in the estimation of the recovered image with exponent $q \in [0, 1]$, which is responsible for controlling the sparsity regulator (sparsity parameter), $\lambda$ is the regularization parameter, $D^{(i)} \in \mathbb{R}^{d \times m}$ is a matrix that was calculated for each output pixel, four directional filters (figure 2).

![Directional filters](image)

Figure 2 – Directional filters used by the regularization term: Horizontal, Diagonal Right, Left Vertical and Diagonal.

C. BID Algorithm [4, 7]

The purpose of BID algorithm involves the minimization of the cost function (3) with respect to the image $x$ and the filter $h$ of the cost function [8, 9]. The image retrieval is oriented to the desired solution, starting with a maximum $\lambda$ (regularization parameter), which is then gradually reduced until it reaches a stopping criterion.

<table>
<thead>
<tr>
<th>Algorithm 1: BID [4] [7]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1- Set $h$ to the identity filter,</td>
</tr>
<tr>
<td>2- Set $x = y$,</td>
</tr>
<tr>
<td>3- Set $\lambda = \lambda_0$,</td>
</tr>
<tr>
<td>repeat</td>
</tr>
<tr>
<td>4- $x \leftarrow \arg \min_x C(x, h)$, with $h$ fixed,</td>
</tr>
<tr>
<td>5- $h \leftarrow \arg \min_h C(x, h)$, with $x$ fixed,</td>
</tr>
<tr>
<td>6- $\lambda_{k+1} = \frac{\lambda_k}{r}$,</td>
</tr>
<tr>
<td>7- until stopping criterion is satisfied.</td>
</tr>
</tbody>
</table>

At the beginning of the optimization, with high value of $\lambda$, only the main features survive in the estimated image. As the optimization progresses, the estimate of the blurring filter improves, and smaller and weaker features can be progressively estimated, leading the method to a good solution. $\lambda_0 = 0.5$ was chosen to be the initial value of the regularization parameter and $r = 2$ was chosen in the geometric progression, step 6 of Algorithm 1.

D. ADMM algorithm

The Algorithm 2 ADMM - Alternating Direction Method of Multipliers is an efficient tool to solve various optimization problems, such as [10] [11]. Consider the general minimization problem:

$$\min_{x \in \mathbb{R}^n} \sum_{j=1}^l g^{(j)}(G^{(j)}x),$$  \hspace{1cm} (5)

where $z \in \mathbb{R}^d$ is the vector to be optimized, $G^{(j)} \in \mathbb{R}^{p_j \times d}$ are arbitrary matrices and $g^{(j)} : \mathbb{R}^{p_j} \rightarrow \mathbb{R}$. Algorithm 2 solves the problem (5) with a variable splitting, which leads to an equivalent problem:

$$\min_{u^{(j)}} \sum_{j=1}^l g^{(j)}(u^{(j)}),$$  \hspace{1cm} (6)

s.t $u^{(j)} = G^{(j)}z$. \hspace{1cm} (7)

The main steps of Algorithm 2, are the lines 4 and 5 whose calculation can be a challenge. Line 4 is a solution of a linear system, while line 5 is a proximity operating function:

$$\text{prox}(y)_{f(x)} = \arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|y - x\|_2^2 + f(x),$$  \hspace{1cm} (8)

which is well known to have a closed shape for several functions $f(x)$ [12].
Algorithm 2: ADMM [5]

1- Set $k = 0$, choose $\mu^{(j)} > 0$, $u_0^{(j)}$ and $d_0^{(j)}$, for $j = 1, \ldots, J$.

repeat

2- $\psi_{k+1} \leftarrow \sum_{j=1}^{J} \mu^{(j)} (G^{(j)})^T (u_{k}^{(j)} + d_{k}^{(j)})$,

3- $z_{k+1} \leftarrow \sum_{j=1}^{J} \mu^{(j)} (G^{(j)})^T G^{(j)} \psi_{k+1}$,

4- $u_{k+1}^{(j)} \leftarrow \text{prox}_{\mu^{(j)}} \left( \frac{G^{(j)} z_{k+1} - d_{k}^{(j)}}{\mu^{(j)}} \right)$,

5- $d_{k+1}^{(j)} \leftarrow d_{k}^{(j)} - \left( G^{(j)} z_{k+1} - u_{k+1}^{(j)} \right)$,

6- $k \leftarrow k + 1$,

7- until stopping criterion is satisfied.

E. Image Estimate with ADMM

Line 4 of the BID optimization algorithm 1 can be written:

$$\min_{u \in \mathbb{R}} \frac{1}{2} \| y - Mu^{(1)} \|^2 + \lambda \sum_{i} \| u^{(2)} \|^q_i .$$

(9)

Equation 9 fits the ADMM formulation of (7) with $G^{(1)} = A$ and $G^{(2)} = D$, the splitting variables $u^{(1)}$ and $u^{(2)}$, and their respective $g^{(j)}$ functions:

$$g^{(1)}(u^{(1)}) = \frac{1}{2} \| y - Mu^{(1)} \|^2 ,$$

(10)

$$g^{(2)}(u^{(2)}) = \lambda \sum_{i} \| u^{(2)} \|^q_i ,$$

(11)

where $D \in \mathbb{R}^{n \times m}$ is a matrix that contains the stacking by lines of $D_{(i)}$ and $u_{(i)} \in \mathbb{R}^d$ is the vector with the output of four directional filters at pixel $i$. Image is estimated using line 3 and 4 from Algorithm 2:

$$z_{k+1} = x_{k+1} = \left( \mu^{(1)} H^T H + \mu^{(2)} D^T D \right)^{-1} \psi_{k+1} ,$$

(12)

$$\psi_{k+1} = \left[ \mu^{(1)} H^T (u_{k}^{(1)} + d_{k}^{(1)}) + \mu^{(2)} D^T (u_{k}^{(2)} + d_{k}^{(2)}) \right] ,$$

(13)

with $D^T$ and $H^T$ obtained from the conjugate of the fast Fourier transform (FFT) of the regularization filters and the deblurring filter respectively, with an inversion operation being easily performed in the transform domain.

The update of the first splitting variable $u_{(k+1)}^{(1)}$, line 5 of Algorithm 2, can be expressed as:

$$u_{(k+1)}^{(1)} = \left( M^T M + \mu^{(1)} \right)^{-1} \left[ M^T y + \mu^{(1)} (H x_{k+1} - d_{(k)}^{(1)}) \right] ,$$

(14)

defining the update of the first splitting variable, where $M^T M$ is a binary matrix that sets to zero all boundary pixels that should not be observed; $M^T y$ is the operation of an extended version of the matrix $y$, with extra pixels equal to zero at the borders. Both operation are hold in the space domain and $H x_{k+1}$ is the operation performed in the transform domain.

The update of the second splitting variable $u_{(k+1)}^{(2)}$ is given by:

$$u_{(k+1)}^{(2)} = \min_{u} g^{(2)}(u^{(2)}) + \mu^{(2)} \| H x_{(k+1)} - u^{(2)} - d^{(2)} \|^2 = \min_u \frac{\lambda}{2} \| u^{(2)} \|^2 + \frac{1}{2} \| D x_{(k+1)} - u^{(2)} - d^{(2)} \|^2 .$$

(15)

The equation (15) is given by:

$$u^{(2)} = \text{vecht} \left( S^{(2)}, \frac{\lambda}{\mu^{(2)}}, q \right) = \text{shrink} \left( S^{(2)}, \frac{\lambda}{\| S^{(2)} \|_2}, q \right) .$$

(16)

where the function $\text{shrink}$ is given by:

$$\text{shrink} \left( S^{(2)}, \lambda, q \right) = \text{prox}_{\| \cdot \|^q} \left( S^{(2)} \right) = \min_t \frac{1}{2} \| S^{(2)} - t \|_2^2 + \lambda \| t \|^q .$$

(17)

For $q = 0.5$, equation (16) has a closed form solution, and can be calculated from the roots of a third order:

$$u_{k+1}^{(2)} = \text{sign} \left( S^{(2)} \right) w ,$$

(18)

where $w$ is given by:

$$w + \frac{1}{2} \lambda w^{-\frac{1}{2}} - \text{abs} \left( S^{(2)} \right) = 0 \Leftrightarrow$$

$$w + \frac{1}{2} \lambda w^{-\frac{1}{2}} - \text{abs} \left( S^{(2)} \right) = 0 \Leftrightarrow$$

$$w^\frac{3}{2} + \frac{1}{2} \lambda - \text{abs} \left( S^{(2)} \right) = 0 .$$

(19)

Making a change of variables $z = \sqrt{w}$ equation (19) can be written as:

$$z^3 + \frac{1}{2} \lambda - \text{abs} \left( S^{(2)} \right) = 0$$

(20)

with $z > 0, w > 0$ .

Finally, variables $d_{(k)}^{(1)}$ and $d_{(k)}^{(2)}$ corresponding to line 6 of Algorithm 2 are updated according to:

$$d_{(k+1)}^{(1)} = d_{(k)}^{(1)} - ( H x_{(k+1)} - u_{(k+1)}^{(1)} ) ,$$

(21)

$$d_{(k+1)}^{(2)} = d_{(k)}^{(2)} - ( D x_{(k+1)} - u_{(k+1)}^{(2)} ) ,$$

(22)

where the operations $H x_{(k+1)}$ and $D x_{(k+1)}$ are easily performed in the transform domain.

F. Filter Estimate with ADMM

Similarly to previous section, line 5 of Algorithm 1 can be written as follows:

$$\min_{h \in \mathbb{R}} \frac{1}{2} \| y - Mu^{(1)} \|^2 + i_5 + u^{(2)} .$$

(23)

Equation 23 fits the ADMM formulation of (7) with $G^{(1)} = A$ and $G^{(2)} = I$, the splitting variables $u^{(1)}$ and $u^{(2)}$, and their respective $g^{(j)}$ functions:
\begin{align*}
g^{(1)}(u^{(1)}) &= \frac{1}{2} ||y - Mu^{(1)}||_2^2, \\
g^{(2)}(u^{(2)}) &= i_5(u^{(2)}).
\end{align*}

The filter update is given by line 3 and 4 of Algorithm 2:
\begin{align*}
z_{k+1} &= h_{k+1} = \left(\mu^{(1)}X^TX + \mu^{(2)}I\right)^{-1}\psi_{k+1},
\psi_{k+1} &= \left[\mu^{(1)}X^T(u_k^{(1)} + d_k^{(1)}) + \mu^{(2)}(u_k^{(2)} + d_k^{(2)})\right],
\end{align*}
where $X^T$ is the matrix obtained from the conjugate of the FFT of the image and $I \in \mathbb{R}^{m \times m}$ is an identity matrix.

For the first splitting variable $u_{k+1}^{(1)}$, line 5 of Algorithm is given by:
\begin{align*}
u_{k+1}^{(1)} &= (M^TM + \mu^{(1)})^{-1}[M^Ty + \mu^{(1)}(Xh_{k+1} - d_k^{(1)})],
\end{align*}
and the second splitting variable is updates as follows:
\begin{align*}
v_{k+1}^{(2)} &= P_S(h_{k+1} - d_k^{(2)}),
\end{align*}
where $P_S(w)$ is the projection of pixels $w$ in the domain of $S^*$, which consists in setting to zero all pixels, that are negative and are that outside the filter support.

Finally, variables $d_1^{(1)}$e $d_2^{(2)}$ corresponding to line 6 of Algorithm 2, are updated according to:
\begin{align*}
d_{k+1}^{(1)} &= d_k^{(1)} - (Xh_{k+1} - u_k^{(1)}),
\end{align*}
\begin{align*}
d_{k+1}^{(2)} &= d_k^{(2)} - (h - u_k^{(2)}),
\end{align*}
where the operation $Xh_{k+1}$ is easily performed in the transform domain.

\section{Blind Image Deblurring With Super-Resolution}

\subsection{Observation Model and Cost Function}

This method was constructed based on the BID method described in section II. That BID method was modifying to perform super-resolution and operate in a multi-frame scenario.

The Super-Resolution degradation model can be formulated as:
\begin{align*}
y_f &= M_fD_fx + n, 
\end{align*}
where $D_f \in \mathbb{R}^{m \times m}$ is the matrix diagonal in frequency corresponding to the degradation of blurred image with the deblurring filter, $y_f \in \mathbb{R}^n$, $x \in \mathbb{R}^m$ and $n \in \mathbb{R}^n$ are column vectors representing, respectively, the degraded image, the original image and additive noise, all vectors ordered lexicographically.

\section{BID and SR Algorithm}

Depending on the number of images and the resolution factor used($\tau$), the method has a different behavior.

Resolution factor decides the mask to be used, which means, if the resolution is equal to 1 is used the previous mask (without decimation), if resolution is greater than 1 is used the new mask with decimation. So we have:
\begin{itemize}
  \item for $f=1$ and resolution=1, this method is equivalent to the BID method in Algorithm 2,
  \item for $f>1$ and resolution=1, is used the Algorithm 3 in a multi-frame scenario,
  \item for $f\geq1$ and resolution>1, is used the Algorithm 3 BID and Super-Resolution.
\end{itemize}

\begin{algorithm}
1- Set $h_f$ to the identity filter,
2- Set $\lambda = \lambda_0$,
3- Set resolution factor,
4- $x \leftarrow \arg \min_{x} C_{SR}(x, h_f)$, with $h_f$ fixed,
5- $h_f \leftarrow \arg \min_{h_f} C_{SR}(x, h_f)$, with $x$ fixed,
6- $\lambda_{k+1} \leftarrow \lambda_k/\tau$,
7- until stopping criterion is satisfied.
\end{algorithm}

\subsection{Image and filter estimate with BID and SR}

In the estimation of the image $x$, we use the above method same adaptations as:
\begin{align*}
\min_{x \in \mathbb{R}^m} \frac{1}{2} ||y_f - M_fu^{(1)}||_2^2 + \lambda [||u^{(2)}||_2^q]
\end{align*}
Equation 35 fits the ADMM formulation of (7) with $G^{(1)} = D_f$ and $G^{(2)} = D$, the splitting variables $u^{(1)}$ and $u^{(2)}$, and their respective $g^{(j)}$ functions:
\begin{align*}
g^{(1)}(u) &= \frac{1}{2} ||y_f - M_fu^{(1)}||_2^2,
\end{align*}
\[ g^{(2)}(u^{(2)}) = \lambda \sum_t \| u^{(2)}_t \|_2^q. \] 

Image is estimated using line 4 Algorithm 3, similar to line 3 and 4 from Algorithm 2:

\[ x^{k+1} = \left( \sum_t \mu^{(1)} D^T D_t + \sum_t \mu^{(2)} D^T D \right)^{-1} \left[ \sum_t \mu^{(1)} D^T (u_k^{(1)} + d_k^{(1)}) + \sum_t \mu^{(2)} D^T (u_k^{(2)} - d_k^{(2)}) \right] \] 

with \( D^T \) and \( D^T \) obtained from the conjugate of the fast Fourier transform (FFT) for the regularization filters and the blurring filter respectively, with the inversion operation performed in the transform domain.

The update of the first splitting variable \( u_k^{(1)} \), line 5 of Algorithm 2, can be expressed as:

\[ u_k^{(1)} = (\mu^{(1)} + M_{ir}^T M_{ir})^{-1} [M_{ir}^T y_k + \mu^{(1)} (d_k^{(1)} - D_k x_{k+1})] \] 

The filter \( h_f \) estimate is the same used in the previous BID method [5], which estimation is made for each filter \( f \).

IV. EXPERIMENTAL RESULTS

The experiments results focus on the methods implemented: the BID method [5] and the proposed extension to perform blind image deblurring with super-resolution in a multi-frame. For each of the following methods the "cameraman" was used as reference image (figure 3), which was compared with each image estimate calculating the improved signal-to-noise-ratio - ISNR or signal-to-noise-ratio - SNR.

All experiments were run with the following parameters: \( q = 0.5, \lambda_0 = 0.5, r = 2 \). The entire process of the methods is controlled by a stopping criterion. Both image and filter estimates are subjected to an outer cycle with a fixed number of iterations (stopping criteria) equal to 40. Image estimation is performed as detailed in section II-E, while filter estimation is performed as detailed in section II-F. The image estimation process and filter estimation process are also controlled by a fixed number of iterations, being 20 and 15 respectively. At the end of the outer cycle, we will have \( n \) image estimates and \( n-1 \) filter estimates, and the best pair is chosen based on the highest value of ISNR or SNR. Blurred Signal-to-Noise Ratio (BSNR) was used for measuring the amount of noise level.

A. BID Method [5]

The experiments done for our implementation of the BID method [5] (algorithm 2), were made with four different deblurring filters, all with a size of 9x9 pixels (figure 4), with and without noise.

![Deblurring filters used from left to right: square, circular, linear motion and non-linear motion.](image)

Table 1 contains the results (ISNR and processing time) obtained with our implementation of the BID method (algorithm 2) for images blurred with different filters and noise levels.

<table>
<thead>
<tr>
<th>Filter</th>
<th>ISNR (dB)</th>
<th>Time</th>
<th>ISNR (dB)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Square</td>
<td>13.34</td>
<td>1min51s</td>
<td>10.05</td>
<td>1min37s</td>
</tr>
<tr>
<td>Circular</td>
<td>9.38</td>
<td>1min43s</td>
<td>8.88</td>
<td>1min35s</td>
</tr>
<tr>
<td>Linear Motion</td>
<td>9.05</td>
<td>1min17s</td>
<td>8.22</td>
<td>1min22s</td>
</tr>
<tr>
<td>Non-linear Motion</td>
<td>11.14</td>
<td>1min18s</td>
<td>8.98</td>
<td>1min17s</td>
</tr>
</tbody>
</table>

Table 1 – Experimental results for our implementation of the BID method [5].

Figures 5 to 8 show the results of the recovered image and the estimated filter obtained for different blurring filter, with and without noise.

![Filter and image estimate obtained for the image degraded with square blurring filter without noise.](image)
This method estimated with quality the image and the blurring filter using different noise levels and different blurring filters.

**B. Blind Image Deblurring in a Multi-frame scenario**

The experiment conducted in this section compares the estimation of a image degraded (single-frame scenario) and nine images in a degrade (multi-frame scenario to the method proposed in section III Algorithm 3).

The blurring filters are the same for both methods that are shown in figure 9. In the single-frame scenario we have one blurred image and in the multi-frame scenario we have nine different blurred images. All blurred images were added additive noise.

Table 2 contains the results for the calculation of SNR (dB) for both experiments in single-frame scenario and multi-frame scenario.

<table>
<thead>
<tr>
<th>Method</th>
<th>BSNR=40</th>
<th></th>
<th>BSNR=20</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SNR(dB)</td>
<td>Time</td>
<td>SNR(dB)</td>
<td>Time</td>
</tr>
<tr>
<td>Single-Frame</td>
<td>19.09</td>
<td>3min11s</td>
<td>18.55</td>
<td>3min07s</td>
</tr>
<tr>
<td>Multi-frame</td>
<td>20.36</td>
<td>9min40s</td>
<td>20.10</td>
<td>4min39s</td>
</tr>
</tbody>
</table>

The Figures 10 and 11 shows the results of the images and filters estimates in a single-frame scenario and a multi-frame scenario.
The proposed method for blind image deblurring in a multi-frame scenario shown to reach better to reconstruction results than with the based method in a single-frame scenario. This method shows that the image estimate suffers little changes with varying noise for blind image deblurring in a multi-frame scenario.

C. **Blind Image Deblurring with Super-Resolution**

Two experiments were performed for this method with resolution equal to 3. One with only one low resolution blurred image with uniform blurring filter with 3x3 pixels and other with nine low resolution blurred images with uniform blurring filters. Both experiences were done without noise.

Figure 12 shows the results for the first experiment with only one low resolution blurred image and its respective blurring filter.

![Figure 12](image12.png)

Figure 12 – Low resolution blurred image and blurring filter.

Figure 13 shows the results, obtained for the first experiment. The image estimate had a SNR of 16.08 dB and the experience took 3 minutes and 33 seconds.

![Figure 13](image13.png)

Figure 13 – Estimated image and filter using uniform blurring filter with 3x3 pixels, without noise.

For the second experience with nine low resolution blurred images was made four different experiments, each one using a uniform blurring filter (3x3, 5x5, 7x7 and 9x9 pixels).

Figure 14 shows an example of the nine low resolution blurred images and Figure 15 shows the layout of the blurring filters used in Figure 14, in a corresponding order. As Figure 15 shows, where the blurring filter were shifting version of the central filter.

![Figure 14 and 15](image14_15.png)

Table 3 contains the results (SNR and execution time) for experiments with different sizes of blurring filters.

<table>
<thead>
<tr>
<th>Uniform filters (size)</th>
<th>SNR (dB)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3x3</td>
<td>20.17</td>
<td>6min09s</td>
</tr>
<tr>
<td>5x5</td>
<td>17.11</td>
<td>6min04s</td>
</tr>
<tr>
<td>7x7</td>
<td>16.75</td>
<td>7min51s</td>
</tr>
<tr>
<td>9x9</td>
<td>15.89</td>
<td>7min50s</td>
</tr>
</tbody>
</table>

Table 3 – Experimental results for super-resolution method with nine blurred images.

Figures 16-19 show the results for the image and filter estimate for the first low resolution blurred image, for each size of blurring filter that was used.
V. CONCLUSION AND FUTURE WORK

The BID method of [5] was successfully implemented and their experimental results are consistent with the expectations. This method estimated with quality the image and the blurring filter using different noise levels and different blurring filters. Furthermore, it has been found that this method recovers the blurred image in a very satisfactory time.

It has also been successfully implemented an extended method of the BID method [5] in order to achieve the blind image deblurring in a multi-frame scenario and super-resolution.

The proposed method for blind image deblurring in a multi-frame scenario shown to reach better to reconstruction results than with the based method in a single-frame scenario.

The proposed method of super-resolution and blind image deblurring achieved better best results when there was access to a greater number of blurred images.

In a second experiment, the proposed method for super-resolution and blind image deblurring was tested using nine low resolution blurred images when varying the size of the blurring filter. This experiment was expected better results in the image and the filter estimate.

For future work, it would be interesting to extend the BID method [5] for estimating color images and also allow extension for the method of super-resolution and blind image deblurring to include higher-resolution color images.

VI. ACKNOWLEDGEMENT

The implementation of the BID method in section II was done in collaboration with my college Sandro Neto.

VII. REFERENCES


