1 Formulation

In this report, we use a lower-case letter $a$ to denote a scalar, a bold lower-case letter $a$ to denote a $2 \times 1$ vector, a normal capital letter $A$ to denote a vector, and a bold capital letter $A$ to denote a matrix. We denote the matrix product as $AB$, where $A \in \mathbb{R}^{n \times m}$ and $B \in \mathbb{R}^m$, and the element-wise product of two vectors as $A \odot B$, where $A, B \in \mathbb{R}^n$.

The task of obstruction removal is to decompose an input sequence $\{I^t\}$ into the background and obstruction components, $I_B$ and $I_O$. We will first derive an algorithm for the more general case of an unknown, spatially varying alpha map, $\alpha$. Let $\alpha$ vary between $0$ and $1$. In this report, we use a lower-case letter $a$ to denote a scalar, a bold lower-case letter $a$ to denote a $2 \times 1$ vector, a normal capital letter $A$ to denote a vector, and a bold capital letter $A$ to denote a matrix. We denote the matrix product as $AB$, where $A \in \mathbb{R}^{n \times m}$ and $B \in \mathbb{R}^m$, and the element-wise product of two vectors as $A \odot B$, where $A, B \in \mathbb{R}^n$.

The task of obstruction removal is to decompose an input sequence $\{I^t\}$ into the background and obstruction components, $I_B$ and $I_O$. We will first derive an algorithm for the more general case of an unknown, spatially varying alpha map, $\alpha$. Let $\alpha$ vary between $0$ and $1$. In this report, we use a lower-case letter $a$ to denote a scalar, a bold lower-case letter $a$ to denote a $2 \times 1$ vector, a normal capital letter $A$ to denote a vector, and a bold capital letter $A$ to denote a matrix. We denote the matrix product as $AB$, where $A \in \mathbb{R}^{n \times m}$ and $B \in \mathbb{R}^m$, and the element-wise product of two vectors as $A \odot B$, where $A, B \in \mathbb{R}^n$.

In this step, we ignore all the terms in Eq. 1 that only consist of $V_O^t$ and $V_B^t$. Ignoring all the terms that only contain $V_B^t$ and $V_B^t$ in Eq. 1, we get:

$$
\begin{align*}
\min_{I_O, I_B, A, \{V_O^t\}, \{V_B^t\}} & \sum_t \|I^t - W(O^t)I_O - W(I_O^t)A \odot W(I_B^t)I_B\|_1 \\
& + \lambda_1 \|\nabla A\|_2^2 + \lambda_2 (\|\nabla I_O\|_1 + \|\nabla I_B\|_1) + \lambda_3 L(I_O, I_B) + \lambda_4 \sum_t \|\nabla V_O^t\|_1 + \|\nabla V_B^t\|_1 \\
\text{Subject to:} & \\
& 0 \leq I_O, I_B, A \leq 1,
\end{align*}
$$

where $I^t, I_O, I_B, A \in \mathbb{R}^n$, temporal index $t \in \{1, 2, ..., T\}$, and $n$ is number of pixels per frame. $V_O, V_B \in \mathbb{R}^{2N \times n}$, and $W(V_B^t)I_B$ is the warped background component at position $x$. Also, recall that we use $\odot$ to denote element-wise product.

2 Optimization

We use an alternating gradient descent method to solve Eq. 1. We first fix the motion fields $\{V_O^t\}$ and $\{V_B^t\}$ and solve for $I_O, I_B, A$, and then fix $I_O, I_B, A$, and solve for $\{V_O^t\}$ and $\{V_B^t\}$. Similar alternating gradient descent approach for joint estimation has been used in video super resolution [1].

2.1 Decomposition step: fix motion fields $\{V_O^t\}$ and $\{V_B^t\}$, and solve for $I_O, I_B, A$.

In this step, we ignore all the terms in Eq. 1 that only consist of $V_O^t$ and $V_B^t$. Ignoring all the terms that only contain $V_B^t$ and $V_B^t$ in Eq. 1, we get:

$$
\begin{align*}
\min_{I_O, I_B, A} & \sum_t \|I^t - W_O^tI_O - W_B^tA \odot W_B^tI_B\|_1 + \lambda_1 \|\nabla A\|_2^2 \\
& + \lambda_2 (\|\nabla I_O\|_1 + \|\nabla I_B\|_1) + \lambda_3 L(I_O, I_B),
\end{align*}
$$

where $\lambda_1$ and $\lambda_2$ are the sets of motion vectors for the obstruction and background components, respectively. According to the Section 4.1 of [2].

2.1 Decomposition step: fix motion fields $\{V_O^t\}$ and $\{V_B^t\}$, and solve for $I_O, I_B, A$.

In this step, we ignore all the terms in Eq. 1 that only consist of $V_O^t$ and $V_B^t$. Ignoring all the terms that only contain $V_B^t$ and $V_B^t$ in Eq. 1, we get:

$$
\begin{align*}
\min_{I_O, I_B, A} & \sum_t \|I^t - W_O^tI_O - W_B^tA \odot W_B^tI_B\|_1 + \lambda_1 \|\nabla A\|_2^2 \\
& + \lambda_2 (\|\nabla I_O\|_1 + \|\nabla I_B\|_1) + \lambda_3 L(I_O, I_B),
\end{align*}
$$

where $\lambda_1$ and $\lambda_2$ are the sets of motion vectors for the obstruction and background components, respectively. According to the Section 4.1 of [2].

2.1 Decomposition step: fix motion fields $\{V_O^t\}$ and $\{V_B^t\}$, and solve for $I_O, I_B, A$.

In this step, we ignore all the terms in Eq. 1 that only consist of $V_O^t$ and $V_B^t$. Ignoring all the terms that only contain $V_B^t$ and $V_B^t$ in Eq. 1, we get:

$$
\begin{align*}
\min_{I_O, I_B, A} & \sum_t \|I^t - W_O^tI_O - W_B^tA \odot W_B^tI_B\|_1 + \lambda_1 \|\nabla A\|_2^2 \\
& + \lambda_2 (\|\nabla I_O\|_1 + \|\nabla I_B\|_1) + \lambda_3 L(I_O, I_B),
\end{align*}
$$

where $\lambda_1$ and $\lambda_2$ are the sets of motion vectors for the obstruction and background components, respectively. According to the Section 4.1 of [2].
First, we approximate the non-smooth L1-norm by the smooth function \( \phi(x) = \sqrt{x^2 + \epsilon^2} \), where \( \epsilon \) is a very small number:

\[
\min_{\lambda O, \hat{I}_B, A} \sum_t \phi(||I^t - W_O^t I_O - W_O^t A \circ W_B^t \hat{I}_B||^2) + \lambda_1 \|
AB\|^2 \\
+ \lambda_2 \left( \phi(||D_x I_O||^2 + ||D_y I_O||^2) + \phi(||D_x \hat{I}_B||^2 + ||D_y \hat{I}_B||^2) \right) + \lambda_3 L(I_O, I_B),
\]

Subject to

\[
0 \leq I_O, I_B, A \leq 1, 
\]

where \( D_x, D_y \in \mathbb{R}^{n \times n} \) are the derivative matrices defined as: \( (D_x I_B)(x) = \frac{\partial I_B}{\partial x} \) and \( (D_y I_B)(x) = \frac{\partial I_B}{\partial y} \) (note that \( D_x I_B \in \mathbb{R}^n \)).

Let \( \hat{I}_O, \hat{I}_B \) and \( \hat{A} \) be the obstruction component, the background component, and the alpha map of the last iteration, respectively. Here we use another approximation used in IRLS:

\[
\phi(x^2) = \sqrt{x^2 + \epsilon^2} \approx \frac{x^2}{\sqrt{x^2 + \epsilon^2}} = \frac{x^2}{\phi(x^2 + \epsilon^2)} \approx \frac{x^2}{\phi(\hat{x}^2 + \epsilon^2)}
\]

where \( \hat{x} \) is very close to \( x \). Eq. 4 can be further simplified as:

\[
\min_{\lambda O, \hat{I}_B, A} \sum_t \frac{\phi(||I^t - W_O^t I_O - W_O^t A \circ W_B^t \hat{I}_B||^2)}{\phi(||I^t - W_O^t I_O - W_O^t A \circ W_B^t \hat{I}_B||^2)} + \lambda_1 \|
AB\|^2 \\
+ \lambda_2 \left( \frac{\phi(||D_x I_O||^2 + ||D_y I_O||^2)}{\phi(||D_x I_O||^2 + ||D_y I_O||^2)} + \frac{\phi(||D_x \hat{I}_B||^2 + ||D_y \hat{I}_B||^2)}{\phi(||D_x \hat{I}_B||^2 + ||D_y \hat{I}_B||^2)} \right) + \lambda_3 L(I_O, I_B),
\]

Subject to

\[
0 \leq I_O, I_B, A \leq 1, 
\]

Note all the denominators are constant in Eq. 6. For simplicity, we denote:

\[
w_1 = \phi(||I^t - W_O^t I_O - W_O^t A \circ W_B^t \hat{I}_B||^2)^{-1}, \quad (6) \\
w_2 = \phi(||D_x \hat{I}_B||^2 + ||D_y \hat{I}_B||^2)^{-1}, \quad (7) \\
w_3 = \phi(||D_x I_O||^2 + ||D_y I_O||^2)^{-1}. \quad (8)
\]

The weights \( w_1, w_2, \) and \( w_3 \) for each pixel are different, so that their vectors, not scalar.

Second, we linearize all the higher-order terms using first-order Taylor expansion. The data term is linearized as ¹

\[
I^t - W_O^t I_O - W_O^t A \circ W_B^t \hat{I}_B = I^t - W_O^t I_O - W_O^t A \circ W_B^t \hat{I}_B - W_O^t A \circ W_B^t \hat{I}_B + W_O^t \hat{A} \circ W_B^t \hat{I}_B \quad (10)
\]

Here we use an approximation commonly used in optimization:

\[
xy \approx x\hat{y} + \hat{x}y - \hat{x}\hat{y},
\]

where \( \hat{x} \) is very close to \( x \) and \( \hat{y} \) is very close to \( y \).

We also linearize the edge ownership term as:

\[
\lambda_3 L(I_O, I_B) = \lambda_3 \sum_x ||\nabla I_O||^2 ||\nabla I_B||^2 \approx \sum_x ||\nabla \hat{I}_O||^2 ||\nabla \hat{I}_B||^2 + ||\nabla I_O||^2 ||\nabla \hat{I}_B||^2 - ||\nabla \hat{I}_O||^2 ||\nabla \hat{I}_B||^2 \\
= \lambda_3 L(I_O, I_B) + L(I_O, \hat{I}_B) - L(\hat{I}_O, \hat{I}_B). \quad (12)
\]

At last, we incorporate the two inequality constraints into our objective function using the penalty method [2]. For example, for the non-negativity constrain \( I_B \geq 0 \), we include the following penalty function into the objective function:

\[
\lambda_p \min(0, I_B)^2 \quad (13)
\]

¹ We make an approximation commonly used in optimization: \( xy \approx x\hat{y} + \hat{x}y - \hat{x}\hat{y} \), where \( \hat{x} \) is very close to \( x \) and \( \hat{y} \) is very close to \( y \).
where \( I_B^2 \) denotes element-wise square and \( \lambda_p \) is the weight for the penalty (we fix \( \lambda_p = 10^5 \)). This function will apply a penalty proportional to the negativity of \( I_B \) (and will be zero if \( I_B \) is nonnegative). Then we add the following terms to enforce all the inequality constraints.

\[
\lambda_p \left( \min(0, I_B)^2 + \min(0, 1 - I_B)^2 + \min(0, I_O)^2 + \min(0, 1 - I_O)^2 + \min(0, A)^2 + \min(0, 1 - A)^2 \right) \tag{14}
\]

Plugging Eq. 10, Eq. 12, and Eq. 14 into Eq. 2, we have:

\[
\min_{I_O, I_B, A} \sum_t w_t \left( \| I_t^\ddagger - W_{I_O} - W_{I_B}^t A \circ W_{I_B}^t \hat{I}_B - W_{I_O}^t \hat{A} \circ W_{I_B}^t \hat{I}_B \| + \| \nabla A \|^2 \right. \\
+ \lambda_2 \left( \| D_x I_O \|^2 + \| D_y I_O \|^2 \right) + \lambda_3 L(I_O, I_B) \\
+ \lambda_p \left( \min(0, I_B)^2 + \min(0, 1 - I_B)^2 + \min(0, I_O)^2 + \min(0, A)^2 + \min(0, 1 - A)^2 \right) \tag{15}
\]

Eq. 15 is a quadratic objective function with respect to \( I_B, I_O, \) and \( A, \) and its minimum is found by solving the linear system which we get by setting the derivative of Eq. 15 to zero. We use Successive over-relaxation (SOR) to solve it [3].

### 2.2 Motion estimation step: fix \( I_O, I_B, A, \) and solve for the motion fields \( V_O^t \) and \( V_B^t \)

When fixing \( I_B \) and \( I_O, \) solving \( v_B \) and \( v_R \) become a joint optical flow problem. Ignoring all the terms that only contain \( I_B \) and \( I_R \) in Eq. 1, we get:

\[
\min_{v_B^t, v_O^t} \| I_t^\ddagger - W(\hat{V}_O^t) - W(\hat{V}_B^t) A \circ W(\hat{V}_B^t) \|_1 + \| \nabla V_O^t \|_1 + \| \nabla V_B^t \|_1 \tag{16}
\]

Note that when we solve the motion field, there is no dependency between frames, so that we can solve each frame independently.

First, we explicitly represent the warping matrices \( W(\hat{V}_B^t) \) and \( W(\hat{V}_O^t) \).

\[
\min_{v_B^t, v_O^t} \sum_x w_1(x) \| I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \|_1 + \lambda_4 \| \nabla V_O^t \|_1 + \lambda_4 \| \nabla V_B^t \|_1 \tag{17}
\]

Similar to Eq. 2 and Eq. 4, we approximate the non-smooth \( l1 \)-norm by the smooth function \( \phi(x) = \sqrt{x + \epsilon^2} \):

\[
\min_{v_B^t, v_O^t} \sum_x w_1(x) \| I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \|^2 + \lambda_4 w_2(x) \| \nabla V_O^t \|^2 + \lambda_4 w_3(x) \| \nabla V_B^t \|^2 \tag{18}
\]

\[
w_1(x) = \frac{1}{\| I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \|^2}, \tag{20}
\]

\[
w_2(x) = \frac{1}{\| \nabla V_O^t \|^2}, \tag{21}
\]

\[
w_3(x) = \frac{1}{\| \nabla V_B^t \|^2}. \tag{22}
\]

where \( \hat{V}_O^t \) and \( \hat{V}_B^t \) be the motion fields of obstruction and background layers of the last iteration, respectively.

Second, we linearize all the higher-order terms using first-order Taylor expansion. The data term is linearized as:

\[
= I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \\
= I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) + A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \\
def I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \\
def I_t(x) - I_O(x - \hat{V}_O^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) - A(x - \hat{V}_O^t(x)) I_B(x - \hat{V}_B^t(x)) \\
- \nabla I_O(x - \hat{V}_O^t(x))^T (\delta V_O^t(x) - \delta V_O^t(x)) \\
- \nabla A(x - \hat{V}_O^t(x))^T (\delta V_O^t(x) - \delta V_O^t(x)) I_B(x - \hat{V}_B^t(x) + \hat{V}_B^t(x)) \tag{23}
\]
where
\[ \overline{A}(x) \triangleq A(x - V^t_O(x)), \quad \overline{I}_O(x) \triangleq I_O(x - \hat{V}^t_O(x)), \quad \overline{A}_B(x) \triangleq I_B(x - \hat{V}^t_B(x)), \quad \nabla I^t_O(x) \triangleq \nabla I_O(x - \hat{V}^t_O(x)), \quad \nabla I^t_B(x) \triangleq \nabla I_B(x - \hat{V}^t_B(x)), \quad \Delta V^t_O = V^t_O - \hat{V}^t_O, \quad \Delta V^t_B = V^t_B - \hat{V}^t_B \]

Plugging Eq. 23 into the Eq. 19, we get a quadratic objective function with respect to \( V^t_O \) and \( V^t_B \):
\[
\min_{v^t_B, v^t_O} \sum_x w_1(x) \| I'(x) - \overline{I}_O(x) - \nabla I^t_O(x)^\top \Delta V^t_O(x) - \overline{A}(x) \nabla I^t_B(x) - \overline{A}_B(x) \nabla \overline{A}(x)^\top \Delta V^t_B(x) \|^2 + \lambda_4 w_2(x) \| \nabla V^t_O(x) \|^2 + \lambda_4 w_3(x) \| \nabla V^t_B(x) \|^2.
\]

Similarly, we find the minimum of Eq. 25 using Successive over-relaxation.

In the case of a reflective pane, the alpha map \( A \) is essentially absorbed into the background and reflection images and there is no need to solve for it separately. We thus remove the prior term \( \| \nabla A \|^2 \) (Eq. 1) from the objective function, and only solve for \( I_O, I_B, \{ V^t_O \}, \) and \( \{ V^t_B \} \).

References

