Real-Time Depth Enhancement by Fusion for RGB-D Cameras

Frederic Garcia*, Djamila Aouada*, Thomas Solignac‡,
Bruno Mirbach‡, and Björn Ottersten*

*Interdisciplinary Centre for Security, Reliability and Trust
Universtity of Luxembourg
{frederic.garcia,djamila.aouada,bjorn.ottersten}@uni.lu

‡Advanced Engineering - IEE S.A.
{thomas.solignac,bruno.mirbach}@iee.lu

Abstract. This paper presents a real-time refinement procedure for depth data acquired by RGB-D cameras. Data from RGB-D cameras suffers from undesired artifacts such as edge inaccuracies or holes due to occlusions or low object remission. In this work, we use recent depth enhancement filters intended for Time-of-Flight cameras, and extend them to structured light based depth cameras, such as the Kinect camera. Thus, given a depth map and its corresponding 2-D image, we correct the depth measurements by separately treating its undesired regions. To that end, we propose specific confidence maps to tackle areas in the scene that require a special treatment. Furthermore, in the case of filtering artifacts, we introduce the use of RGB images as guidance images as an alternative to real-time state-of-the-art fusion filters that use grayscale guidance images. Our experimental results show that the proposed fusion filter provides dense depth maps with corrected erroneous or invalid depth measurements and adjusted depth edges. In addition, we propose a mathematical formulation that enables to use the filter in real-time applications.

Key words: depth enhancement, data fusion, active sensing.

1 Introduction

The research on autonomous systems that are capable of understanding the shape and location of objects in a scene has been growing in recent years. Hence the demand for a quality depth estimation is today one of the active research areas in computer vision. Recent advances in active sensor technologiesby the hand of PrimeSense™have greatly helped to significantly overcome this problem, and consumer-accessible RGB-D cameras such as the Kinect distributed by Microsoft® , the Xtion Pro Live distributed by Asus, or the Carmine distributed and produced by the same manufacturer PrimeSense™, are able to

This work was supported by the National Research Fund, Luxembourg, under the CORE project C11/BM/1204105/FAVE/Ottersten.

provide high-resolution depth maps in real-time. However, such sensing systems estimate depth from triangulation techniques and thus, they are linked to the baseline between the camera and the light source, which yields to occlusions or shadowing, and creates erroneous regions during depth estimation.

Similar non-desired artifacts have been tackled in stereopsis; *e.g.*, large occluded regions can be handled by image in-painting [1] techniques that adopt structure propagation and texture synthesis. The extension of such approaches to active sensing systems is known as hole filling and is commonly used in depth image based rendering for 3-D TV [2, 3]. However, its performance is far from real-time. We, instead base our work on approaches of fusion by filtering tested and proven on Time-of-Flight (ToF) cameras [4–7]. Our main goal is to generalize such approaches and define a novel fusion filter, to which we refer as the RGB-D filter, specifically intended for real-time RGB-D consumer cameras. The filtering is based on the concept of fusing a given depth map with a guidance or a reference image (or images), usually taken as the matching 2-D image. In general, in the case of fusion filters for ToF depth enhancement, the guidance image is used to upsample the low resolution depth maps to their same resolution. In our case, instead, this guidance image is used to correct unreliable depth regions.
In this paper, we will design new confidence measures to incorporate to the proposed filter in order to indicate those areas within the initial depth map that require special attention.

The remainder of the paper is organized as follows: Section 2 presents the general framework of depth resolution enhancement by fusion filters. In Section 3, we present the RGB-D filter as well as the confidence measures to combine depth and 2-D information. Section 4 proposes a mathematical formulation for the RGB-D filter that enables for a real-time performance. In Section 5, we present and quantify the results of the proposed depth enhancement approach. Finally, concluding remarks are given in Section 6.

2 Problem Statement & Background

The idea of considering a guidance 2-D image to improve the quality of its corresponding depth map was first introduced by Kopf *et. al* in [8], where they presented the Joint Bilateral Upsampling (JBU) filter, an extension of the bilateral filter [9] that considers two different data sources within the kernel of the filter. Their work was first intended to compute a solution for image analysis and enhancement tasks, such as tone mapping or colorization through a downsampled version of the data. However, its application to depth data enhancement inspired most of the current depth enhancement techniques by fusion [4–6, 10]. The JBU filter enhances an initial depth map $\mathbf{D}$ to the higher resolution of a corresponding 2-D guidance image $\mathbf{I}$, as follows

$$\mathbf{J_1}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)}, \tag{1}$$

where $N(\mathbf{p})$ is the neighborhood at the pixel indexed by the position vector $\mathbf{p} = (i, j)^T$, with i and j indicating the row, respectively column corresponding to the pixel position. This non-iterative filter formulation is a weighted average of the local neighborhood samples, where the weights are computed based on spatial and radiometric distances between the center of the considered sample and the neighboring samples. Thus, its kernel is decomposed into a spatial weighting term $f_{\mathbf{S}}(\cdot)$ that applies to the pixel position $\mathbf{p}$, and a range weighting term $f_{\mathbf{I}}(\cdot)$ that applies to the pixel intensity value $\mathbf{I}(\mathbf{q})$. The weighting functions $f_{\mathbf{S}}(\cdot)$ and $f_{\mathbf{I}}(\cdot)$ are generally chosen to be Gaussian functions with standard deviations $\sigma_{\mathbf{S}}$ and $\sigma_{\mathbf{I}}$, respectively. Nevertheless, according to the bilateral filter principle, the fundamental heuristic assumptions about the relationship between depth and intensity data, may lead to erroneous copying of 2-D texture into actually smooth geometries within the depth map. Furthermore, a second unwanted artifact known as edge blurring appears along depth edges that have no corresponding edges in the 2-D image, *i.e.*, in situations where objects on either side of a depth discontinuity have a similar color. In order to cope with these issues, we proposed in [5] a new fusion filter known as Pixel Weighted Average Strategy (PWAS). The PWAS filter extends the expression in (1) by an additional factor, to which we referred as the credibility map, that indicates unreliable regions within the depth maps obtained using a Time-of-Flight (ToF) camera. Since these unreliable regions are linked to depth edges, we defined the credibility map as

$$\mathbf{Q}(\mathbf{p}) = \exp\left(\frac{-\big(\nabla\mathbf{D}(\mathbf{p})\big)^2}{2\sigma_{\mathbf{Q}}^2}\right), \tag{2}$$

where $\nabla\mathbf{D}$ is the gradient of the given depth map $\mathbf{D}$. From (2), depth pixels in $\mathbf{D}$ that belong to smooth regions, *i.e.*, no depth gradient, are weighted with a high value ($\mathbf{Q} \approx 1$) to indicate that measurements in these pixels have a high reliability. In contrast, depth pixels that belong to depth edges are weighted with a low value ($\mathbf{Q} \approx 0$) to indicate that these pixels are not reliable. We then focus on these low reliable depth pixels as they require a special treatment. Thus, for a given depth map $\mathbf{D}$, a credibility map $\mathbf{Q}$, and a guiding intensity image $\mathbf{I}$, the enhanced depth map $\mathbf{J_2}$ resulting from PWAS filtering is defined as follows

$$\mathbf{J_2}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}(\mathbf{p}),\mathbf{I}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})}. \tag{3}$$

Although, the PWAS filter copes well with edge blurring, texture copying is still not fully solved within the enhanced depth maps. The reason is that the range weighting term within the filter in (3) considers a textured guidance image. In order to significantly reduce this artifact, we proposed in [6] the Unified Multi-Lateral (UML) filter. The UML filter combines two PWAS filters where the output $\mathbf{J_3}$ of the second one has both spatial and range weighting terms acting onto the same data source $\mathbf{D}$. In addition, we suggested to use the credibility map $\mathbf{Q}$ as a blending function, *i.e.*, $\beta = \mathbf{Q}$, hence, depth pixels with high reliability

are not influenced by the 2-D data avoiding texture copying as follows

$$\mathbf{J_4}(\mathbf{p}) = \big(1 - \beta(\mathbf{p})\big) \cdot \mathbf{J_2}(\mathbf{p}) + \beta(\mathbf{p}) \cdot \mathbf{J_3}(\mathbf{p}), \tag{4}$$

with

$$\mathbf{J_3}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{D}}\big(\mathbf{D}(\mathbf{p}),\mathbf{D}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{D}}\big(\mathbf{D}(\mathbf{p}),\mathbf{D}(\mathbf{q})\big)\mathbf{Q}(\mathbf{q})}. \tag{5}$$

The kernel $f_{\mathbf{D}}$ is another Gaussian function with a different standard deviation $\sigma_{\mathbf{D}}$.

3 Proposed RGB-D Filter for Depth Enhancement

In the following, we demonstrate how fusion filters designed to enhance the accuracy and the resolution of low-resolution depth maps from ToF cameras can be extended to enhance the data given by consumer RGB-D cameras such as the Kinect or the Xtion Pro Live. In contrast to ToF cameras, RGB-D cameras estimate depth using structured light techniques and thus, problems arise when the light power of the projected pattern is not sufficient to be reflected back to the sensor. Furthermore, since active triangulation methods require a baseline for depth estimation, occlusion is an additional drawback to overcome. We herein consider a consumer RGB-D camera as an active sensor that provides a depth map $\mathbf{D}$ and a perfectly matching 2-D color image $\mathbf{I}$. Depth maps obtained from such RGB-D cameras have no depth information within occluded regions. Moreover, depth measurements can be unreliable in object boundaries. We note that in the case of the PWAS or UML filters, non-valid depth measurements were treated separately in order to be preserved in the enhanced depth map, *e.g.*, background pixels [11]. In the case of RGB-D cameras, non-valid pixels are set to 0. Since the aim of this paper is to fill these regions with appropriate depth measurements, we propose to replace the credibility map $\mathbf{Q}$ in $\mathbf{J_2}$ by $\mathbf{Q_D}$, defined as

$$\mathbf{Q_D}(\mathbf{p}) = \begin{cases} 0 & \text{if } \mathbf{D}(\mathbf{p}) = 0, \\ \mathbf{Q}(\mathbf{p}) & \text{otherwise.} \end{cases} \tag{6}$$

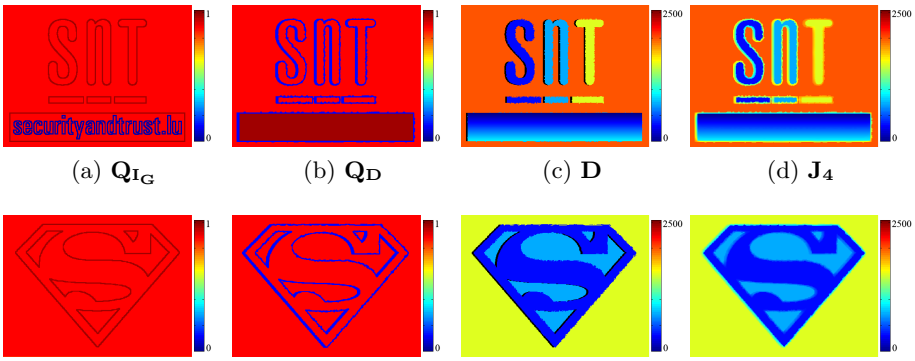
We recall that pixels with low weights in $\mathbf{Q_D}$ indicate a low reliable depth pixel while pixels with high weights indicate a reliable depth pixel.

The use of the credibility map $\mathbf{Q}$ as a blending function β, suggested in (4), provokes edge blurring when filtering low reliable depth pixels if no corresponding 2-D edge is present. Although this situation only arises when foreground and background objects share the same intensity value, the conversion of the given color image $\mathbf{I}$ to its grayscale version $\mathbf{I_G}$, that is used as a guidance image, increases this unwanted effect. Indeed, different colors in the RGB space get collapsed to the same grayscale intensity value. Fig. 1 presents two synthetic images that make this effect more prominent, illustrating the loss of 2-D edges when transforming to grayscale. Thus, in order to reduce edge blurring in the filter output, we proposed in [12] to generalize the blending function β in (4) as follows

$$\beta(\mathbf{p}) = \mathbf{Q_D}(\mathbf{p})\Big(1 + \mathbf{Q_{I_G}}(\mathbf{p})\big(1 - \mathbf{Q_D}(\mathbf{p})\big)\Big), \tag{7}$$

(a) $\mathbf{I}$ (b) $\mathbf{I_G}$ (c) $\mathbf{I}$ (d) $\mathbf{I_G}$

Fig. 1: (a) First test case. (b) Grayscale conversion of (a). (c) Second test case. (d) Grayscale conversion of (c).



(a) $\mathbf{Q_{I_G}}$ (b) $\mathbf{Q_D}$ (c) $\mathbf{D}$ (d) $\mathbf{J_4}$

(e) $\mathbf{Q_{I_G}}$ (f) $\mathbf{Q_D}$ (g) $\mathbf{D}$ (h) $\mathbf{J_4}$

Fig. 2: 1^{st} col.: $\mathbf{Q_{I_G}}$ computed on Fig. 1b and Fig. 1d. 2^{nd} col.: $\mathbf{Q_D}$ computed on (c) and (d). 3^{rd} col.: Depth maps $\mathbf{D}$. 4^{rd} col.: Filter responses $\mathbf{J_4}$.

with the factor $\mathbf{Q_{I_G}}$ defined analogously to $\mathbf{Q}$ in (2) but considering $\nabla\mathbf{I_G}$, the gradient of $\mathbf{I_G}$. By doing so, we give more weight to the PWAS term $\mathbf{J_2}$ in the case where $\mathbf{Q_{I_G}}$ is low, *i.e.*, if there exists a 2-D edge to which it is possible to adjust the unreliable depth measurements. However, in the case where no 2-D edge is present, *i.e.*, $\mathbf{Q_{I_G}} \approx 1$, inaccurate depth measurements along depth edges cannot be aligned. Fig. 2a and Fig. 2e are the confidence measures $\mathbf{Q_{I_G}}$ computed on the two test cases in Fig. 1b and Fig. 1d, respectively. We can observe that in both cases, $\mathbf{Q_{I_G}}$ is high for all non reliable depth pixels ($\mathbf{Q_{I_G}} \approx 0$). As a result, the filter outputs $\mathbf{J_4}$ shown in Fig. 2d and Fig. 2h, respectively, present edge blurring within object boundaries.

In general, RGB-D cameras include an RGB sensor intended for display purposes, *e.g.*, Microsoft® uses it to display what the RGB-D camera sees. In this paper, we propose to take advantage of this sensor and to use its data as colour guidance images. By doing so, we overcome the aforementioned problem when transforming from RGB to grayscale. However, when real-time is required, the straightforward usage of RGB images as three channels to guide the filtering becomes impractical. If we consider an RGB image within the range kernel in (3),

its dimensionality increase becomes intractable due to memory restrictions [13]. A possible solution would be to consider each RGB channel separately and then to combine the three filter outputs. However, this combination may also lead to edge blurring since edges might not be visible in all three channels, as demonstrated in [13, 14]. Instead, we propose to filter considering each RGB channel adaptively, *i.e.*,

$$\mathbf{J_2}_{,c}(\mathbf{p}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}_c(\mathbf{p}),\mathbf{I}_c(\mathbf{q})\big)\mathbf{Q_D}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(\mathbf{I}_c(\mathbf{p}),\mathbf{I}_c(\mathbf{q})\big)\mathbf{Q_D}(\mathbf{q})}. \tag{8}$$

The subscript c indicates the channel of the RGB image $\mathbf{I} = \{\mathbf{I}_c\}, c \in \{0,1,2\}$ to be filtered. Our aim is to consider for each pixel $\mathbf{p}$ the RGB channel $c(\mathbf{p})$ that best describes a corresponding 2-D edge when filtering; that is, to choose the RGB channel in which the edge is visible to better guide the filter (see Fig. 3). To that end, we define three new confidence measures $\mathbf{Q}_{\mathbf{I}_c}$ to quantify the presence of a 2-D edge for each RGB channel. $\mathbf{Q}_{\mathbf{I}_c}$ is defined analogously to $\mathbf{Q}_{\mathbf{I_G}}$ but replacing $\mathbf{I_G}$ by $\mathbf{I}_c$. The RGB-D filter is then defined as follows

$$\mathbf{J_5}(\mathbf{p}) = \big(1-\beta(\mathbf{p})\big)\cdot \mathbf{J_2}_{,c(\mathbf{p})}(\mathbf{p}) + \beta(\mathbf{p})\cdot \mathbf{J_3}(\mathbf{p}), \tag{9}$$

with

$$\beta(\mathbf{p}) = \mathbf{Q_D}(\mathbf{p})\Big(1+\mathbf{Q_I}(\mathbf{p})\big(1-\mathbf{Q_D}(\mathbf{p})\big)\Big), \tag{10}$$

and

$$[\mathbf{Q_I}(\mathbf{p}), c(\mathbf{p})] = \min\{\mathbf{Q}_{\mathbf{I}_c}(\mathbf{p})\}, \quad c \in \{0,1,2\}. \tag{11}$$

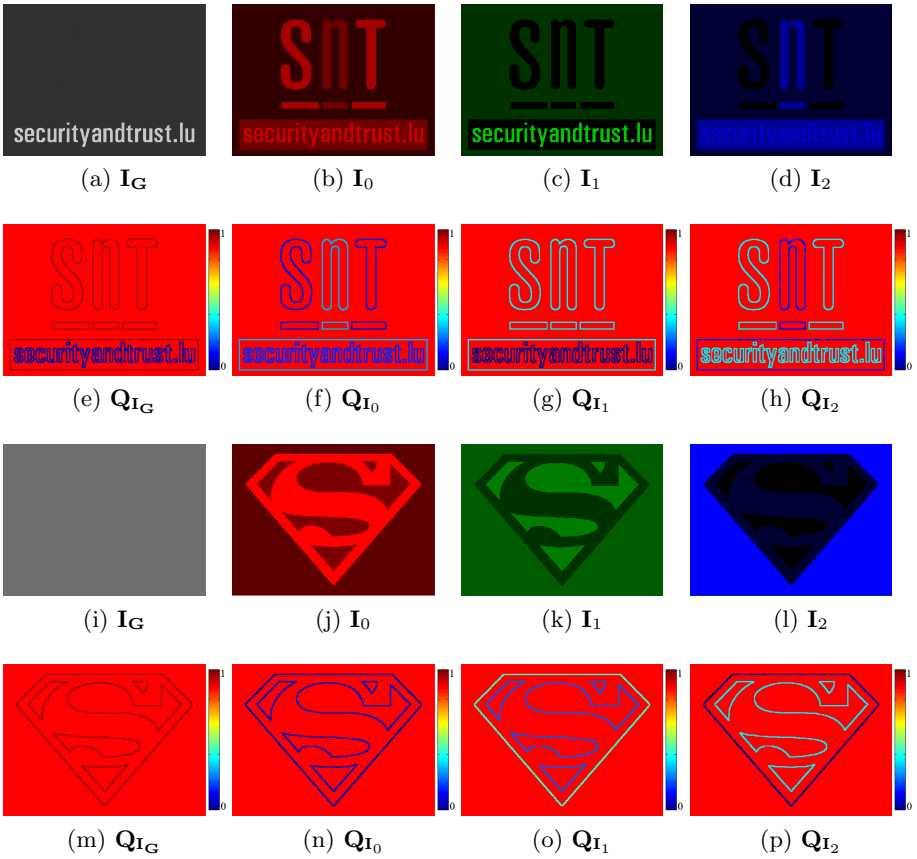
The function $\min(\mathbf{A})$ returns the smallest element along different dimensions of an array $\mathbf{A}$, and its corresponding index. We recall that $\mathbf{J_2}_{,c(\mathbf{p})}$ is the PWAS filter output that considers as a guidance image the RGB channel $\mathbf{I}_{c(\mathbf{p})}$ that best describes the 2-D edge for each pixel $\mathbf{p}$, as shown in Fig 4.

4 Real-Time Implementation

Though bilateral filtering is known to be time consuming, its latest fast implementations based on data quantization and downsampling [13, 15], enable a high-performance. Inspired on these techniques, we proposed a real-time implementation for the PWAS and UML filters in [6]. In the following, we extend our previous work and we propose an RGB-D filter formulation intended for real-time performance.

4.1 Range Data Quantization

Similarly to [15], we sample the range of the 2-D intensity values and depth measurements, *i.e.*, $I_{c,k} = s_{\mathbf{I}} \cdot k$, and $D_l = s_{\mathbf{D}} \cdot l$, with $k = 0, ..., K$ and $l = 0, ..., L$. $s_{\mathbf{I}}$ and $s_{\mathbf{D}}$ are the 2-D and depth quantization factors; thus $(s_{\mathbf{I}} \times K)$ and $(s_{\mathbf{D}} \times L)$ are equal or larger than the maximum 2-D intensity values and depth



(a) $\mathbf{I_G}$ (b) $\mathbf{I}_0$ (c) $\mathbf{I}_1$ (d) $\mathbf{I}_2$

(e) $\mathbf{Q_{I_G}}$ (f) $\mathbf{Q_{I_0}}$ (g) $\mathbf{Q_{I_1}}$ (h) $\mathbf{Q_{I_2}}$

(i) $\mathbf{I_G}$ (j) $\mathbf{I}_0$ (k) $\mathbf{I}_1$ (l) $\mathbf{I}_2$

(m) $\mathbf{Q_{I_G}}$ (n) $\mathbf{Q_{I_0}}$ (o) $\mathbf{Q_{I_1}}$ (p) $\mathbf{Q_{I_2}}$

Fig. 3: 1^{st} and 3^{rd} rows: 2-D guidance images. 2^{nd} and 4^{th} rows: 2-D confidence measures.

measurements, respectively. Then, inserting in (8) and in (5) the quantized levels $I_{c,k}$ and D_l for $\mathbf{I}_c(\mathbf{p})$, respectively $\mathbf{D}(\mathbf{p})$, one obtains for each intensity channel and for each level a filtered range image

$$\mathbf{J_2}_{,c}(\mathbf{p}, I_{c,k}) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(I_{c,k}, \mathbf{I}_c(\mathbf{q})\big)\mathbf{Q_D}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{I}}\big(I_{c,k}, \mathbf{I}_c(\mathbf{q})\big)\mathbf{Q_D}(\mathbf{q})}, \tag{12}$$

and

$$\mathbf{J_3}(\mathbf{p}, D_l) = \frac{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big)\mathbf{Q_D}(\mathbf{q})\mathbf{D}(\mathbf{q})}{\sum_{\mathbf{q}\in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p},\mathbf{q}) f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big)\mathbf{Q_D}(\mathbf{q})}. \tag{13}$$

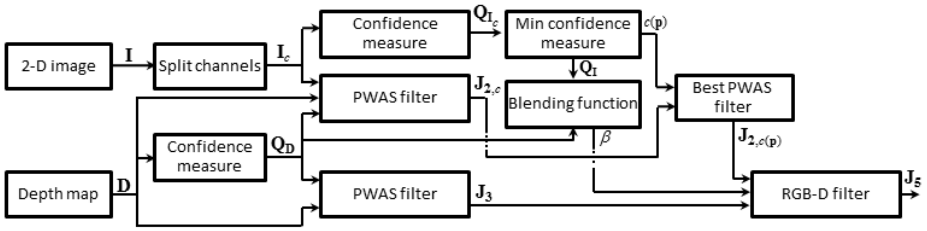


Fig. 4: Flow diagram of the RGB-D filter computation. The subscript c stands for the channels of $\mathbf{I}$, *i.e.*, $c \in \{0, 1, 2\}$ and $c(\mathbf{p})$ indicates the best channel c to adjust a given pixel $\mathbf{p}$.

We define four mappings, *i.e.*, $E^{I_{c,k}}(\cdot)$ and $F^{I_{c,k}}(\cdot)$, for a quantized intensity value at the pixel position $\mathbf{p}$ such that

$$E^{I_{c,k}} : \mathbf{q} \longmapsto f_{\mathbf{I}}\big(I_{c,k}, \mathbf{I}_c(\mathbf{q})\big) \cdot \mathbf{Q_D}(\mathbf{q}) \cdot \mathbf{D}(\mathbf{q}), \tag{14}$$

$$F^{I_{c,k}} : \mathbf{q} \longmapsto f_{\mathbf{I}}\big(I_{c,k}, \mathbf{I}_c(\mathbf{q})\big) \cdot \mathbf{Q_D}(\mathbf{q}), \tag{15}$$

and $G^{D_l}(\cdot)$ and $H^{D_l}(\cdot)$, for a quantized depth measurement at the pixel position $\mathbf{p}$ such that

$$G^{D_l} : \mathbf{q} \longmapsto f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big) \cdot \mathbf{Q_D}(\mathbf{q}) \cdot \mathbf{D}(\mathbf{q}), \tag{16}$$

$$H^{D_l} : \mathbf{q} \longmapsto f_{\mathbf{D}}\big(D_l, \mathbf{D}(\mathbf{q})\big) \cdot \mathbf{Q_D}(\mathbf{q}). \tag{17}$$

We then may rewrite (12) and (13) as follows

$$\mathbf{J_2}_{,c}(\mathbf{p}, I_{c,k}) = \frac{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) \cdot E^{I_{c,k}}(\mathbf{q})}{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) \cdot F^{I_{c,k}}(\mathbf{q})}, \tag{18}$$

and

$$\mathbf{J_3}(\mathbf{p}, D_l) = \frac{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) \cdot G^{D_l}(\mathbf{q})}{\sum_{\mathbf{q} \in N(\mathbf{p})} f_{\mathbf{S}}(\mathbf{p}, \mathbf{q}) \cdot H^{D_l}(\mathbf{q})}. \tag{19}$$

We note that $f_{\mathbf{S}}(\mathbf{p}, \mathbf{q})$ is a function of the difference $(\mathbf{p} - \mathbf{q})$. We may hence write (18) and (19) as

$$\mathbf{J_2}_{,c}(\mathbf{p}, I_{c,k}) = \frac{\big(f_{\mathbf{S}} * E^{I_{c,k}}\big)(\mathbf{p})}{\big(f_{\mathbf{S}} * F^{I_{c,k}}\big)(\mathbf{p})}, \tag{20}$$

and

$$\mathbf{J_3}(\mathbf{p}, D_l) = \frac{\big(f_{\mathbf{S}} * G^{D_l}\big)(\mathbf{p})}{\big(f_{\mathbf{S}} * H^{D_l}\big)(\mathbf{p})}, \tag{21}$$

where $*$ denotes the convolution between functions.

The filtered value $\mathbf{J_2}_{,c}\big(\mathbf{p},\mathbf{I}_c(\mathbf{p})\big)$ results from a linear interpolation of the filtered range image $\mathbf{J_2}_{,c}(\mathbf{p},\cdot)$ obtained for the different levels at position $\mathbf{p}$ and intensity value $\mathbf{I}_c(\mathbf{p})$ between $I_{c,k}$ and $I_{c,k+1}$, *i.e.*,

$$\begin{aligned}\mathbf{J_2}_{,c}\big(\mathbf{p},\mathbf{I}_c(\mathbf{p})\big) &= \text{interpolate}\big(\mathbf{J_2}_{,c}(\mathbf{p},\cdot),\mathbf{I}_c(\mathbf{p})\big)\\ &= \frac{1}{s_{\mathbf{I}}}\Big(\big(I_{c,k+1}-\mathbf{I}_c(\mathbf{p})\big)\mathbf{J_2}_{,c}(\mathbf{p},I_{c,k+1}) + \big(\mathbf{I}_c(\mathbf{p})-I_{c,k}\big)\mathbf{J_2}_{,c}(\mathbf{p},I_{c,k})\Big).\end{aligned} \tag{22}$$

The same applies to $\mathbf{J_3}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big)$; thus from a linear interpolation between D_l and D_{l+1}, *i.e.*,

$$\begin{aligned}\mathbf{J_3}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big) &= \text{interpolate}\big(\mathbf{J_3}(\mathbf{p},\cdot),\mathbf{D}(\mathbf{p})\big)\\ &= \frac{1}{s_{\mathbf{D}}}\Big(\big(D_{l+1}-\mathbf{D}(\mathbf{p})\big)\mathbf{J_3}(\mathbf{p},D_{l+1}) + \big(\mathbf{D}(\mathbf{p})-D_l\big)\mathbf{J_3}(\mathbf{p},D_l)\Big).\end{aligned} \tag{23}$$

Finally, the enhanced depth map $\mathbf{J_5}$ results from (9), considering the best PWAS filter output $\mathbf{J_2}_{,c(\mathbf{p})}$ from (22), the filtered depth map $\mathbf{J_3}$ from (23), and the blending function β from (10).

4.2 Data Downsampling

In addition to the range quantification, one can ensure a good memory and speed performance by downsampling the data to be filtered. According to the study that Paris et al. conducted in [13], the sampling of the input data does not introduce significant errors. The same strategy applies to the proposed RGB-D filter presented in Section 3. To that end, we downsample the input data, *i.e.*, $\mathbf{I}_{c\downarrow} = \text{downsample}(\mathbf{I}_c,\lambda)$, $\mathbf{D}_{\downarrow} = \text{downsample}(\mathbf{D},\lambda)$, and $\mathbf{Q}_{\mathbf{D}\downarrow} = \text{downsample}(\mathbf{Q}_{\mathbf{D}},\lambda)$, with λ being the scale factor. We run equations (12)-(23) using $\mathbf{I}_{c\downarrow}$ and $\mathbf{D}_{\downarrow}$, instead of $\mathbf{I}_c$, respectively $\mathbf{D}$, from which result four low-resolution filtered images $\mathbf{J_2}_{,c\downarrow}$, and $\mathbf{J_3}_{\downarrow}$. Formally, the values $\mathbf{J_2}_{,c}\big(\mathbf{p},\mathbf{I}_c(\mathbf{p})\big)$ and $\mathbf{J_3}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big)$ of the high-resolution filtered range images can be obtained by spatially interpolating the low-resolution filtered images, *i.e.*,

$$\mathbf{J_2}_{,c}\big(\mathbf{p},\mathbf{I}_c(\mathbf{p})\big) = \text{interpolate}\Big(\mathbf{J_2}_{,c\downarrow}\big(\cdot,\mathbf{I}_c(\mathbf{p})\big),\mathbf{p}/\lambda\Big), \tag{24}$$

and

$$\mathbf{J_3}\big(\mathbf{p},\mathbf{D}(\mathbf{p})\big) = \text{interpolate}\Big(\mathbf{J_3}_{\downarrow}\big(\cdot,\mathbf{D}(\mathbf{p})\big),\mathbf{p}/\lambda\Big). \tag{25}$$

In addition, we propose to combine both the linear range interpolation and the bi-linear spatial interpolation to a tri-linear (*i.e.*, eight point) interpolation as follows

$$\mathbf{J_2}_{,c}\big(\mathbf{p},\mathbf{I}_c(\mathbf{p})\big) = \text{interpolate}\big(\mathbf{J_2}_{,c\downarrow}(\cdot,\cdot),\mathbf{p}/\lambda,\mathbf{I}_c(\mathbf{p})\big), \tag{26}$$

and

$$\mathbf{J_3}\big(\mathbf{p}, \mathbf{D}(\mathbf{p})\big) = \text{interpolate}\big(\mathbf{J_{3\downarrow}}(\cdot,\cdot), \mathbf{p}/\lambda, \mathbf{D}(\mathbf{p})\big). \tag{27}$$

Notice that within these interpolations, the low resolution filtered images $\mathbf{J_{2},}_{c\downarrow}$ have to be computed for each value $\mathbf{I}_c(\mathbf{p})$ of the high resolution input images. Instead, we propose to not compute for each channel c the filter output $\mathbf{J_{2},}_{c}$, but only compute for each pixel $\mathbf{p}$ the filter output of the channel $c(\mathbf{p})$, *i.e.*,

$$\mathbf{J_{2},}_{c(\mathbf{p})}\big(\mathbf{p}, \mathbf{I}_{c(\mathbf{p})}(\mathbf{p})\big) = \text{interpolate}\big(\mathbf{J_{2},}_{c(p)\downarrow}(\cdot,\cdot), \mathbf{p}/\lambda, \mathbf{I}_{c(\mathbf{p})}(\mathbf{p})\big). \tag{28}$$

$c(\mathbf{p})$ being the index of the channel given by (11). The final output of the RGB-D filter is then obtained according to (9) by superposing the filter outputs in (27) and (28) using the blending function β in (10) that defines a pixel-dependent weight for each of the two contributions.

In order to avoid filtering artefacts due to the data quantization and sampling introduced above, the standard deviations $\sigma_{\mathbf{I}}$, $\sigma_{\mathbf{D}}$, and $\sigma_{\mathbf{S}}$ may be chosen greater than $s_{\mathbf{I}}$, $s_{\mathbf{D}}$, and $s_{\mathbf{S}}$, respectively. Otherwise, the approximation may be poor, *i.e.*, numerically unstable. According to the mappings in (15) and (17), the noise due to quantization only affects the range mapping functions $F^{I_{c,k}}$ and H^{D_l}, and thus, the intensity values of the 2-D image $\mathbf{I}_c(\mathbf{q})$ as well as the depth measurements of the depth map $\mathbf{D}(\mathbf{q})$ are preserved.

Through the experiments, we noticed that depth measurements that belong to reliable regions ($\mathbf{D}(\mathbf{p})$ with $\mathbf{Q_D}(\mathbf{p}) \approx 1$) do not present significant noise. This, in turn, allowed us to further optimise the computation time as we could replace the second PWAS filtering $\mathbf{J_3}$ in (9), intended to smooth depth measurements in such regions, by the depth map $\mathbf{D}$ given by the RGB-D camera, *i.e.*,

$$\mathbf{J_6}(\mathbf{p}) = \big(1 - \beta(\mathbf{p})\big) \cdot \mathbf{J_{2},}_{c(\mathbf{p})}(\mathbf{p}) + \beta(\mathbf{p}) \cdot \mathbf{D}(\mathbf{p}), \tag{29}$$

By doing so, we further improve the global time consuming of the RGB-D filter as we avoid the consumption time required to compute $\mathbf{J_3}$. We note that this further time optimization can only be done in case the noise within depth measurements that belong to reliable regions can be neglected.

5 Experimental Results

We herein consider the Xtion Pro Live as a consumer RGB-D camera, however, either the Xtion Pro Live, the Carmine, or the Kinect cameras present similar specifications as all of them are produced by the same manufactured, PrimeSense$^{\text{TM}}$. In addition to be a consumer-accessible depth camera, a major advantages is that the aforementioned depth cameras are a close sensing system that provide real-time depth and video streams with a well done mapping between the integrated sensors. Therefore, we do not have to deal with the internal transformation, in which distance-dependence disparity is involve [16], to map the depth data onto the 2-D data. In the following, we first motivate the use of a

Table 1: Percentage of best SSIM performance per enhanced depth pixel in $\mathbf{J_6}$ when filtering using $\mathbf{I_G}$, $\mathbf{I}$, $\mathbf{I}_0$, $\mathbf{I}_1$, and $\mathbf{I}_2$ as a guidance image.

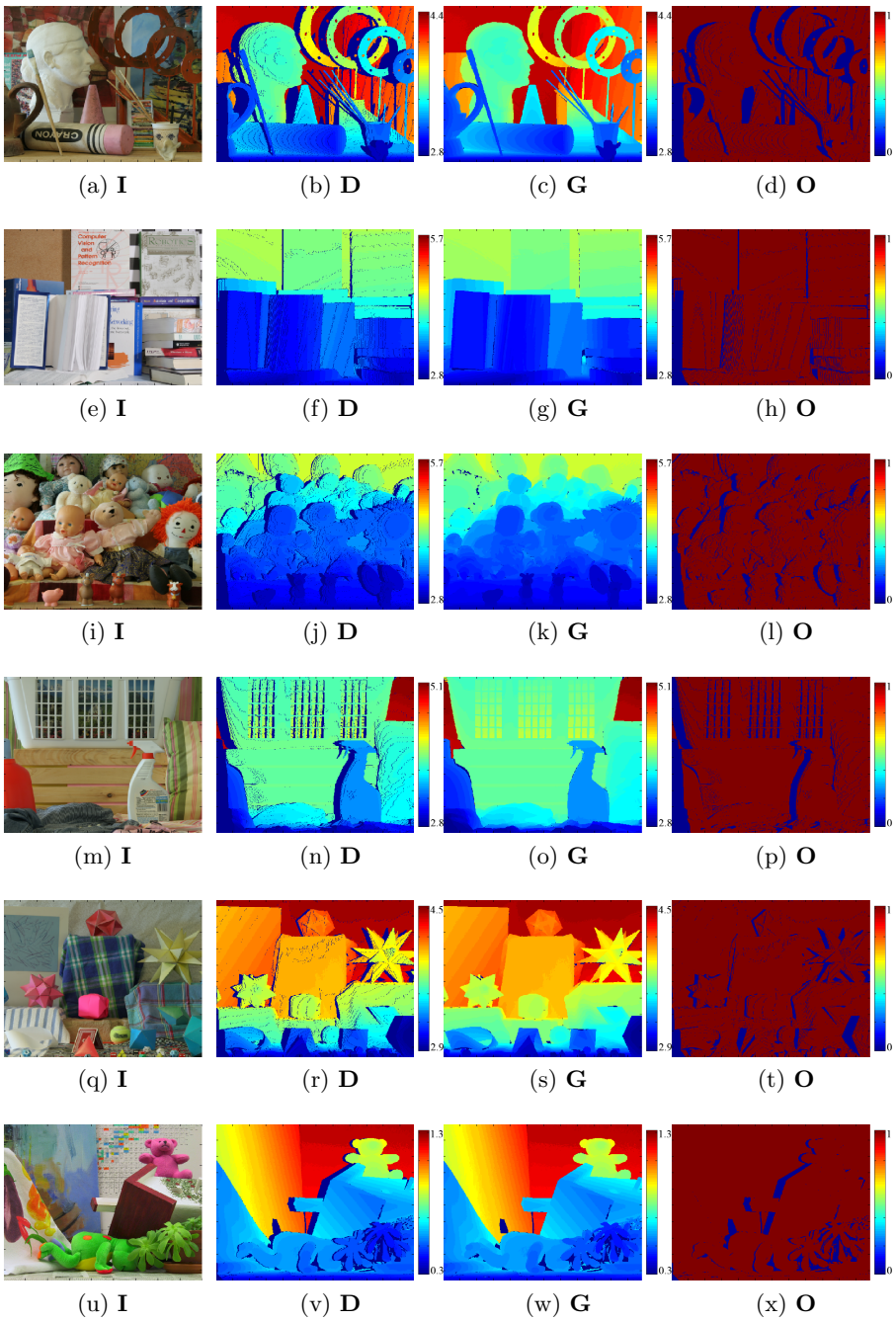
Dataset	$\mathbf{I_G}$	$\mathbf{I}$	$\mathbf{I}_0$	$\mathbf{I}_1$	$\mathbf{I}_2$
Art	19.41	26.02	27.81	20.55	37.33
Books	25.66	39.98	33.77	30.41	38.05
Dolls	15.89	23.84	30.50	19.54	29.64
Laundry	26.46	36.45	34.47	29.26	38.99
Moebius	27.12	38.28	34.86	29.89	35.98
Teddy	43.87	48.13	46.20	52.06	46.12

colour guidance image by a quantitative evaluation in which we have considered scenes from the Middlebury stereo dataset[1] as well as the two test cases that we have previously generated to illustrate the downside of transforming a colour guidance image to its grayscale version. Next, we show some visual results using lab-acquired sequences recorded by the Xtion Pro Live camera. Finally, we perform a runtime analysis at different data sampling rates.

5.1 Qualitative and Quantitative Evaluation

In order to motivate the use of a colour guidance image within the filtering process, we have first quantified the performance of the RGB-D filter using the Art, Books, Dolls, Laundry, Moebius, and Teddy scenes from the Middlebury stereo dataset (see Fig. 5). The Middlebury stereo dataset provides ground truth disparity maps and their corresponding 2-D RGB images. The ground truths $\mathbf{G}$ (see 3^{rd} column of Fig. 5) have been generated from the given ground truth disparity maps using the sensing system parameters from the Middlebury website (focal length is 3740 pixels and baseline is 160 mm). The depth maps to be enhanced $\mathbf{D}$ (see 2^{nd} column of Fig. 5) result from the product between the occlusion masks $\mathbf{O}$ (see 4^{th} column of Fig. 5) and their corresponding ground truth depth maps $\mathbf{G}$. The occlusion masks $\mathbf{O}$ have been generated by crosschecking the pair of given disparity maps. Table 1 presents the qualitative evaluation of the proposed RGB-D filter. We have used the Structural SIMilarity (SSIM) measure [17, 18] as an evaluation metric. Each cell in Table 1 contains the percentage of best SSIM performance per enhanced depth pixel in $\mathbf{J_6}$ when filtering using $\mathbf{I_G}$, $\mathbf{I}$, and $\mathbf{I}_c$ as a guidance image. We recall that $\mathbf{I_G}$ is the grayscale version of $\mathbf{I}$ while the subscript c relates to each of the channels of $\mathbf{I}$, *i.e.*, $c \in \{0, 1, 2\}$, indicating the red, green, and blue channels, respectively. We note that the accuracy of an enhanced depth pixel in $\mathbf{J_6}$ is always higher when filtering using either the red, green, or blue channels of $\mathbf{I}$ than when using $\mathbf{I_G}$. Indeed, a 2-D edge may not be present in all three channels, provoking a low intensity contrast when the three channels are linearly combined to get $\mathbf{I_G}$, *i.e.*, $\mathbf{I_G} = 0.299\ \mathbf{I}_0 + 0.587\ \mathbf{I}_1 + 0.114\ \mathbf{I}_2$.

[1] Middlebury Stereo Dataset, http://vision.middlebury.edu/stereo



(a) **I** (b) **D** (c) **G** (d) **O**

(e) **I** (f) **D** (g) **G** (h) **O**

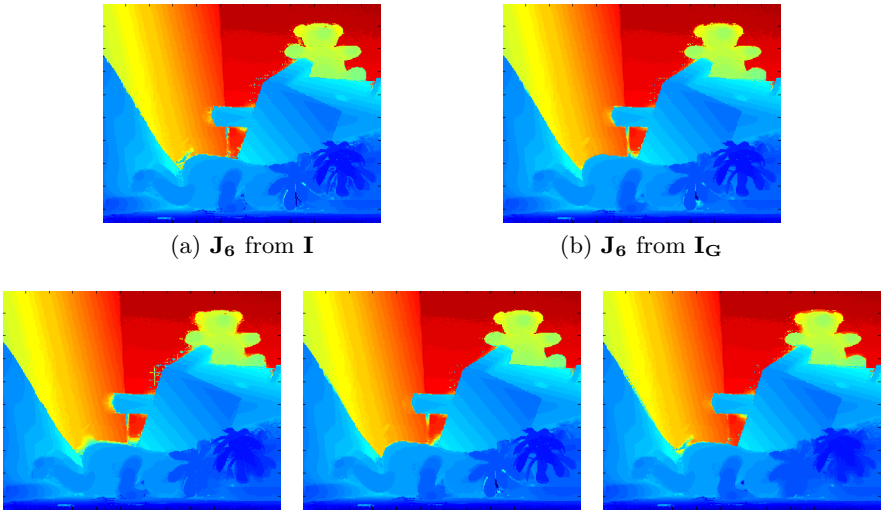
(i) **I** (j) **D** (k) **G** (l) **O**

(m) **I** (n) **D** (o) **G** (p) **O**

(q) **I** (r) **D** (s) **G** (t) **O**

(u) **I** (v) **D** (w) **G** (x) **O**

Fig. 5: 1^{st} row. Art scene. 2^{nd} row. Books scene. 3^{rd} row. Dolls scene. 4^{th} row. Laundry scene. 5^{th} row. Moebius scene. 6^{th} row. Teddy scene. 1^{st} col. RGB guidance image. 2^{nd} col. Depth map to be enhanced. 3^{rd} col. Ground truth. 4^{th} col. Occlusion mask. Depth units are in meters.



(a) $\mathbf{J_6}$ from $\mathbf{I}$ (b) $\mathbf{J_6}$ from $\mathbf{I_G}$

(c) $\mathbf{J_6}$ from $\mathbf{I}_0$ (d) $\mathbf{J_6}$ from $\mathbf{I}_1$ (e) $\mathbf{J_6}$ from $\mathbf{I}_2$

Fig. 6: RGB-D filter responses using $\mathbf{I}$, $\mathbf{I_G}$, and $\mathbf{I}_c$ guidance images, respectively ($\sigma_{\mathbf{S}} = 10$, $\sigma_{\mathbf{D}} = 10$, $\sigma_{\mathbf{Q_D}} = 100$, $\sigma_{\mathbf{Q_I}} = \sigma_{\mathbf{D}}$, $s_{\mathbf{I}} = 50$, $s_{\mathbf{D}} = 300$, $\lambda = 8$).

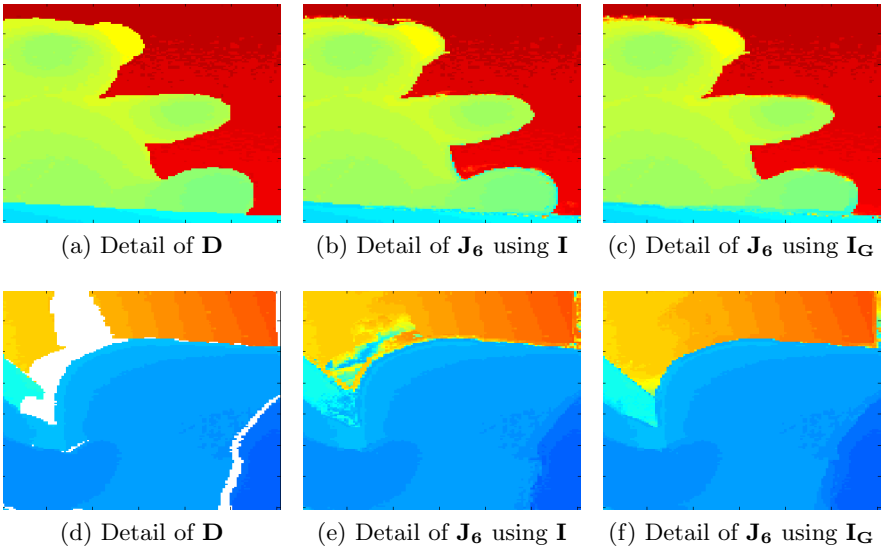
Table 2: Quantitative evaluation using the SSIM evaluation metric on a detail of the RGB-D filter responses presented in Fig. 6.

Row in Fig. 7	$\mathbf{J_6}$ from $\mathbf{I}$	$\mathbf{J_6}$ from $\mathbf{I_G}$	$\mathbf{J_6}$ from $\mathbf{I}_0$	$\mathbf{J_6}$ from $\mathbf{I}_1$	$\mathbf{J_6}$ from $\mathbf{I}_2$
1st row	**87.70**	86.32	83.48	89.15	79.75
2nd row	95.06	**96.50**	91.92	97.51	92.77

Table 3: SSIM evaluation of the proposed RGB-D filter presented on Fig. 6 (Non-valid pixels in $\mathbf{G}$ are not considered).

$\mathbf{D}$	$\mathbf{J_6}$ using $\mathbf{I}$	$\mathbf{J_6}$ using $\mathbf{I_G}$	$\mathbf{J_6}$ using $\mathbf{I}_0$	$\mathbf{J_6}$ using $\mathbf{I}_1$	$\mathbf{J_6}$ using $\mathbf{I}_2$
84.95	94.20	**94.61**	93.25	95.00	93.06

We next quantify the RGB-D filter against our previous works. To that end, we have considered the Teddy scene (see last row of Fig. 5). In Fig. 6, we show the RGB-D filter outputs using $\mathbf{I}$, $\mathbf{I_G}$, and $\mathbf{I}_c$ as guidance images, respectively. We observe a much better performance when adjusting depth edges using RGB guidance images (see a detail of the object boundaries in Fig. 7 and the quantification in the first row of Table 2). However, in this specific case, the performance within occluded regions is lower if we use the RGB guidance image than if we use its grayscale version, as presented in the second row of Table 2). According to the proposed β in (10), $\mathbf{Q_I}$ will be linked to the RGB channel that has a higher gradient. In this case, a higher gradient does not mean a better 2-D edge



(a) Detail of $\mathbf{D}$ (b) Detail of $\mathbf{J_6}$ using $\mathbf{I}$ (c) Detail of $\mathbf{J_6}$ using $\mathbf{I_G}$

(d) Detail of $\mathbf{D}$ (e) Detail of $\mathbf{J_6}$ using $\mathbf{I}$ (f) Detail of $\mathbf{J_6}$ using $\mathbf{I_G}$

Fig. 7: Detail of the RGB-D filter responses in Fig. 6.

description but a higher amount of texture that is copied within the low reliable regions indicated by the credibility map $\mathbf{Q_D}$ (see Fig. 8). We note that different weights are assigned to each RGB channel during their linear combination to grayscale conversion. Indeed, a higher weight is assigned to the green channel, which, according to Table 1 and Table 3, produces the best performance. That is why in this specific case, the use of a grayscale guidance image presents a higher performance.

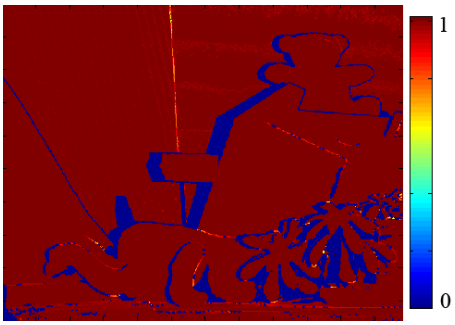
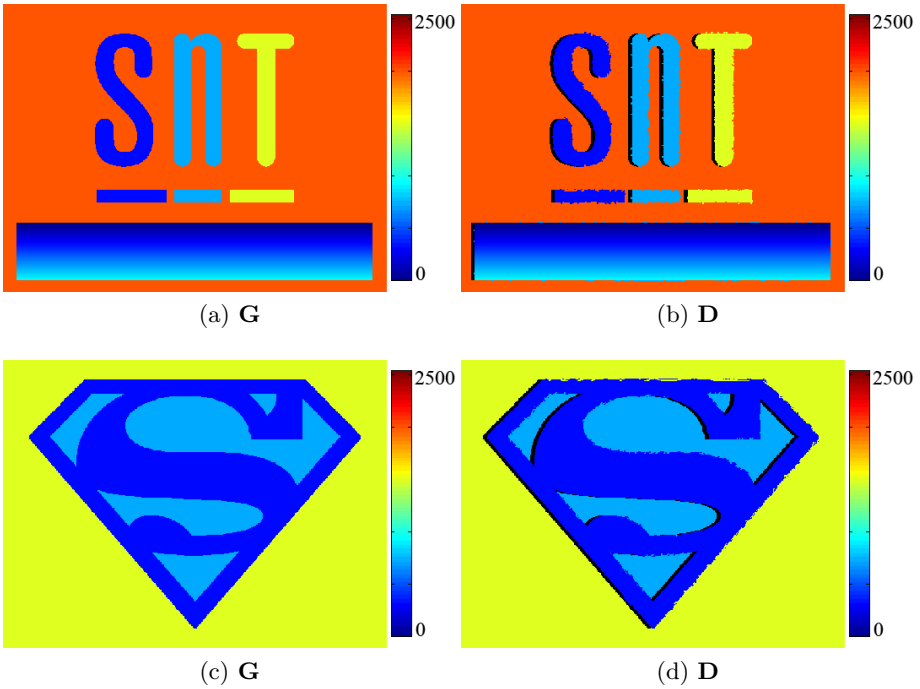


Fig. 8: Credibility map $\mathbf{Q_D}$ of $\mathbf{D}$ in Fig. 5v.

We next evaluate the evolution of our research work to deal with accurate depth data enhancement by fusion. We note that further evaluations and com-

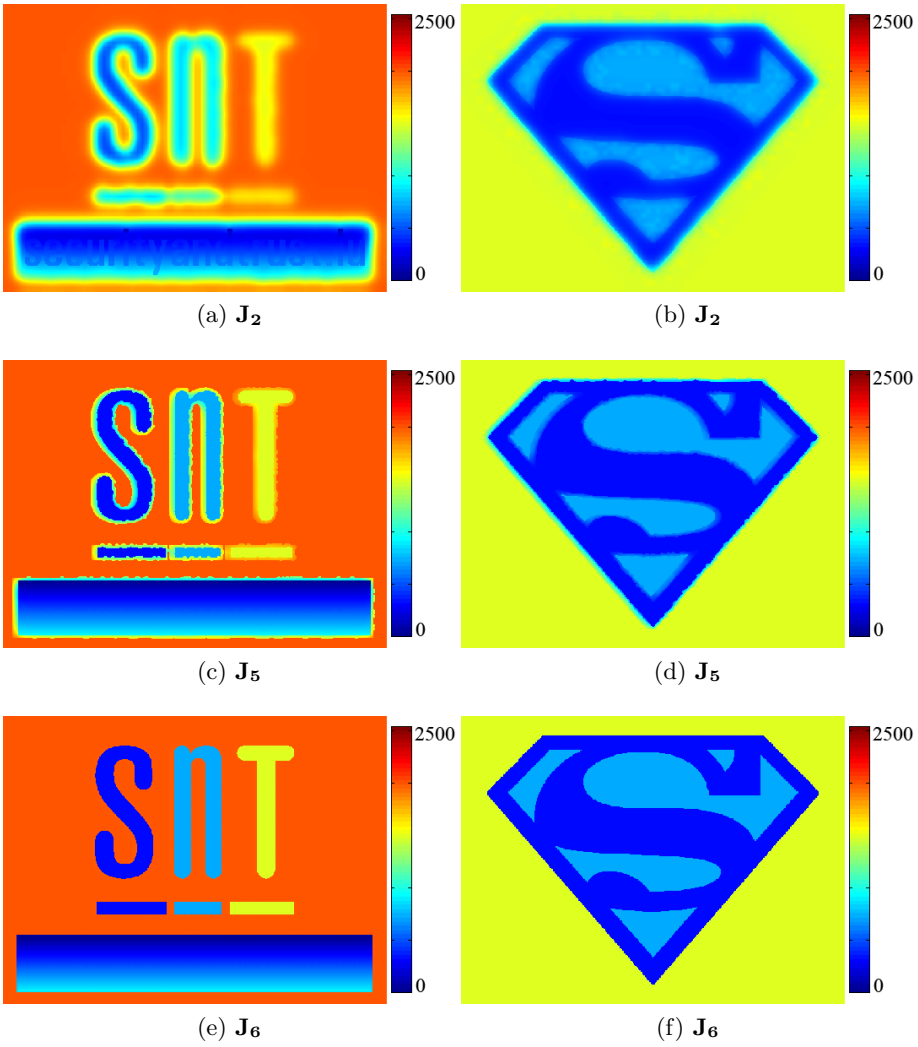


(a) **G** (b) **D**

(c) **G** (d) **D**

Fig. 9: 1^{st} row: Test case 1. 2^{nd} row: Test case 2. Depth units are in millimetres.

parisons of our previous work against state-of-the-art depth enhancement techniques were done in [5, 6], in which we presented an overall improvement in all test cases. However, depth enhancement approaches often fail when 2-D guidance information is not available. To show that, we have considered the two synthetic cases presented in Fig. 1. Their respective ground truth depth maps **G** are shown in Fig. 9a, respectively Fig. 9c, to which we compare the filter outputs using the SSIM measure. Fig. 9b and Fig. 9d are the depth maps **D** to be enhanced, which differ from their corresponding ground truths **G** as they present depth inaccuracies along object boundaries as well as occlusion areas. Fig. 10 presents the enhanced depth maps when using the PWAS filter (1^{st} row), the UML filter (2^{nd} row), and the RGB-D filter (3^{rd} row). We observe that the output of the PWAS filter not only presents edge blurring as a consequence of filtering using grayscale guidance images $\mathbf{I_G}$, but also texture copying. The last artifact is well handled by the UML filter, but edge blurring still occurs. When 2-D edges get suppressed during the transformation from RGB to grayscale, the RGB-D filter gives the best enhanced depth maps. Table 4 presents a comparison between the enhanced depth maps presented in Fig. 10 and their respective ground truth **G**. In both test cases, enhanced depth maps using the RGB-D filter are almost



(a) $\mathbf{J_2}$ (b) $\mathbf{J_2}$

(c) $\mathbf{J_5}$ (d) $\mathbf{J_5}$

(e) $\mathbf{J_6}$ (f) $\mathbf{J_6}$

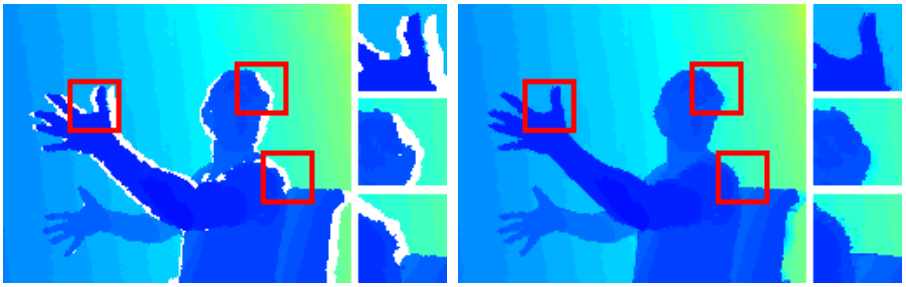
Fig. 10: 1^{st} row. Output of PWAS filter. 2^{nd} row. Output of UML filter. 3^{rd} row. Output of RGB-D filter. Depth units are in millimetres.

equal to the ground truth, which indicates a perfect handling of occlusion areas as well as inaccurate depth measurements along object boundaries.

Finally, we evaluate our filter in a real scene. Fig. 11 shows a raw depth map $\mathbf{D}$ acquired using the Xtion Pro Live camera and its enhanced version $\mathbf{J_6}$ using the proposed filter. The credibility map $\mathbf{Q_D}$ in Fig. 12 indicates those invalid or occluded depth pixels ($\mathbf{Q_D} \approx 0$) in the given depth map $\mathbf{D}$ that have been satisfactorily handled, *i.e.*, replaced by correct depth measurements from their

Table 4: Quantitative evaluation using the SSIM measure between the filter outputs presented in Fig. 10 and their corresponding ground truth depth maps $\mathbf{G}$ (100 is the highest similarity).

	$\mathbf{D}$	$\mathbf{J_2}$	$\mathbf{J_5}$	$\mathbf{J_6}$
1^{st} test case	88.43	53.38	79.66	99.89
2^{nd} test case	86.89	70.62	86.13	99.99

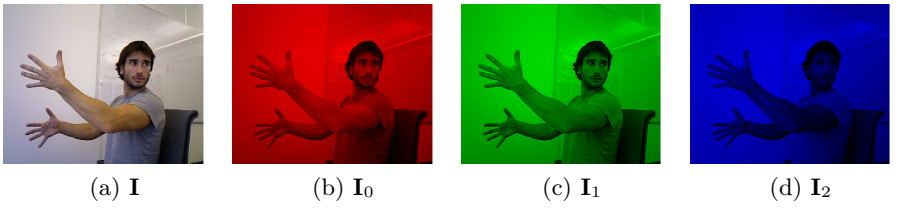


(a) $\mathbf{D}$ (b) $\mathbf{J_6}$

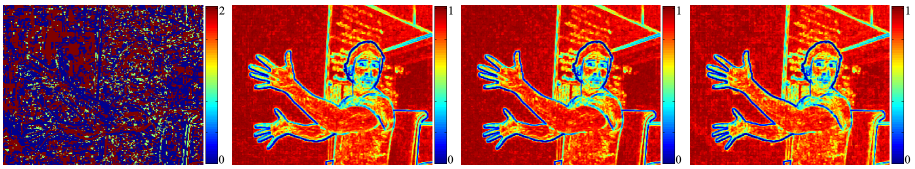
Fig. 11: (a) Raw depth map given by the Xtion Pro Live camera. White areas indicate non-valid (occluded) pixels. (b) Enhanced depth map using the RGB-D filter ($\sigma_{\mathbf{S}} = 10$, $\sigma_{\mathbf{D}} = 10$, $\sigma_{\mathbf{Q_D}} = 100$, $\sigma_{\mathbf{Q_I}} = \sigma_{\mathbf{D}}$, $s_{\mathbf{I}} = 50$, $s_{\mathbf{D}} = 300$, $\lambda = 8$).

Fig. 12: Credibility map $\mathbf{Q_D}$ of the depth map $\mathbf{D}$ in Fig. 11a.

neighbourhood, as shown in Fig. 11b. Depth measurements inaccuracies in object boundaries have been fixed by aligning depth edges to their corresponding ones in the RGB image $\mathbf{I}$, shown in Fig. 13a. Red, green, and blue pixels in Fig. 13e indicate which channel has been chosen when filtering those pixels. This decision was taken using the confidence measures $\mathbf{Q_{I_c}}$ shown in Fig. 13f-Fig. 13h.



(a) $\mathbf{I}$ (b) $\mathbf{I}_0$ (c) $\mathbf{I}_1$ (d) $\mathbf{I}_2$



(e) $c(\mathbf{p})$ (f) $\mathbf{Q}_{\mathbf{I}_0}$ (g) $\mathbf{Q}_{\mathbf{I}_1}$ (h) $\mathbf{Q}_{\mathbf{I}_2}$

Fig. 13: (a) Input RGB image. (b)-(d) are the red, green, and blue channels of $\mathbf{I}$ in (a). The value of each pixel in (e) indicates the RGB channel to be considered for each pixel. (f), (g), and (h) are the confidence measures that describe the 2-D edges for each RGB channel.

5.2 Runtime Analysis

Next, we present a runtime analysis to show that the implementation proposed in Section 4 enables real-time applications. We have implemented the RGB-D filter in C++ language. However, we note that our implementation is not optimized and that many tasks within the filtering process can run in parallel, which would significantly increase its performance. A GPU implementation would also significantly reduce the filtering consumption time. We ran the experiments on a laptop with Intel® Core™i7-2640M CPU @ 2.80GHz with 4 GB of RAM.

Table 5 presents the consumption time to obtain the output of an RGB-D filter for four different scale factors $\lambda \in \{0, 1, 2, 3, 4\}$. That is, we sample the input data at 1x, 2x, 4x, 8x, and 16x, respectively. We provide the SSIM measure to illustrate the induced error linked to each sampling rate. A sampling rate of 8x ($\lambda = 3$) seems to be a good trade-off between quality and speed. We also compare the difference on performance when considering $\mathbf{I}$ and $\mathbf{I}_\mathbf{G}$ as guidance images. As expected, the filtering process is almost three times slower when processing three channels for small sampling rates while this difference is reduced for bigger sampling rates.

6 Conclusions

In this paper, we presented a new fusion filter to remove undesired artifacts from depth data given by consumer depth cameras such as the Kinect, the Carmine, or the Xtion Pro Live. Our main contribution is in identifying the areas of erroneous measurements that require a special treatment such as object boundaries.

Table 5: SSIM measure and runtime analysis of the RGB-D filter using $\mathbf{I}$ and $\mathbf{I_G}$ as a guidance images, and at different sampling rates (time units are in seconds; average over 100 iterations).

Sampling	**SSIM**(100 - 0)		Time consumption (s)	
	Using $\mathbf{I}$	Using $\mathbf{I_G}$	Using $\mathbf{I}$	Using $\mathbf{I_G}$
0x	96.23	96.22	7.34	2.42
2x	96.17	96.18	1.70	0.59
4x	96.06	96.10	0.55	0.20
8x	95.67	95.79	0.30	0.12
16x	94.84	95.17	0.21	0.09

In contrast to ToF cameras, consumer RGB-D cameras present accurate and less noisy depth measurements. This, in turn, allowed us to avoid the depth smoothing operation performed by the second term of the RGB-D filter. That is, we preserve the real sensed data in areas of high depth reliability. Moreover, we propose to tackle the limitations of grayscale images by adaptively using color channels as guidance images where the adaptivity comes from considering the color channel with the highest gradient. The selection of the best channel descriptor when filling occluded areas needs to be further investigated as the current approach tends to preserve 2-D texture within occluded regions.

A real-time formulation has been proposed and evaluated in the experimental part in which we propose to quantize and sample the data to be filtered. A sampling factor of 8x presents a good trade-off between the quality of the enhanced depth map and its computation time. This new formulation enables the use of colour guidance images within the filtering process, which was impractical for real-time applications.

Current applications in which consumer RGB-D cameras are being used, *e.g.*, games, gesture recognition, or human sensing, can benefit from the proposed depth enhancement approach to improve their performance.

References

1. Bin, S., Wei, H., Yimin, Z., Yu-Jin, Z.: Image inpainting via sparse representation. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). (2009)
2. Solh, M., AlRegib, G.: Hierarchical Hole-Filling(HHF): Depth image based rendering without depth map filtering for 3D-TV. In: IEEE International Workshop on Multimedia Signal Processing (MMSP). (2010)
3. Yu-Cheng, F., Tsung-Chen, C.: The Novel Non-Hole-Filling Approach of Depth Image Based Rendering. In: 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video. (2008)
4. Chan, D., Buisman, H., Theobalt, C., Thrun, S.: A noise-aware filter for real-time depth upsampling. In: Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (ECCVW). (2008)

5. Garcia, F., Mirbach, B., Ottersten, B., Grandidier, F., Cuesta, A.: Pixel Weighted Average Strategy for Depth Sensor Data Fusion. In: International Conference on Image Processing (ICIP). (2010) 2805–2808
6. Garcia, F., Aouada, D., Mirbach, B., Solignac, T., Ottersten, B.: A New Multi-lateral Filter for Real-Time Depth Enhancement. In: Advanced Video and Signal-Based Surveillance (AVSS). (2011)
7. Min, D., Lu, J., Minh, N.D.: Depth Video Enhancement Based on Weighted Mode Filtering. IEEE Transactions on Image Processing (TIP) **21** (2012) 1176–1190
8. Kopf, J., Cohen, M., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. In: SIGGRAPH '07: ACM SIGGRAPH 2007 papers, New York, NY, USA, ACM (2007) 96
9. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: ICCV. (1998) 839–846
10. Crabb, R., Tracey, C., Puranik, A., Davis, J.: Real-time foreground segmentation via range and color imaging. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). (2008) 1–5
11. Garcia, F.: Sensor Fusion Combining 3-D and 2-D for Depth Data Enhancement. PhD thesis, The Faculty of Sciences, Technology and Communication. University of Luxembourg (2012)
12. Garcia, F., Aouada, D., Abdella, H.K., Solignac, T., Mirbach, B., Ottersten, B.: Depth Enhancement by Fusion for Passive and Active Sensing. In: 2.5D Sensing Technologies in Motion: the Quest for 3D (QU3ST) in conjunction with European Conference on Computer Vision (ECCV). (2012)
13. Paris, S., Durand, F.: A fast approximation of the bilateral filter using a signal processing approach. In: International Journal of Computer Vision. Volume 81., Kluwer Academic Publishers (2009) 24–52
14. Garcia, F., Aouada, D., Mirbach, B., Ottersten, B.: A New 1-D Colour Model and its Application to Image Filtering. In: International Symposium on Image and Signal Processing and Analysis (ISPA). (2011) 1–4
15. Yang, Q., Tan, K.H., Ahuja, N.: Real-time O(1) bilateral filtering. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). (2009) 557–564
16. Garcia, F., Aouada, D., Mirbach, B., Ottersten, B.: Real-Time Distance-Dependent Mapping for a Hybrid ToF Multi-Camera Rig. IEEE Journal of Selected Topics in Signal Processing **6** (2012)
17. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision **47** (2002) 7–42
18. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. In: IEEE TIP. Volume 13–4. (2004) 600–612