DEMOSAICING LOW RESOLUTION QVGA BAYER PATTERN

Tommaso Guseo and Emanuele Menegatti
Intelligent Autonomous Systems Laboratory (IAS-Lab), The University of Padua

Keywords: Robot vision, demosaicing, QVGA resolution, Bayer pattern.

Abstract: In this paper, we present a solution for the interpolation of low resolution digital images. Many digital cameras can function in two resolution modes: VGA (i.e., 640×480) and QVGA (i.e., 320×240). These cameras use a single sensor covered with a Color Filter Array (CFA). The CFA allows only one color component to be measured at each pixel, the remaining color components must be interpolated, this operation is called demosaicing. There is not a standard way to interpolate the QVGA Bayer pattern and most of the known demosaicing algorithms are not suitable. In this paper, we propose a new solution for the interpolation of QVGA Bayer pattern. Experimental results using digital images and an evaluation function confirm the effectiveness of the interpolation method. The use of the QVGA resolution is important in low-cost and low-power embedded hardware. As an application, we chose the RoboCup domain and in particular our Robovie-M humanoid robot competing in the RoboCup Kid-Size Humanoids League.

1 INTRODUCTION

Color digital cameras are popular today. They can be found also in PDA cellular phone, etc. CMOS digital cameras are more and more often used in autonomous robots to acquire images of the environment surrounding a robot. Most cameras use a single light sensor covered with a Color Filter Array (CFA). The CFA is a colored filter that allows only one color component to be measured at each pixel, the remaining color components must be interpolated: this operation is called demosaicing. Several patterns exist for the CFA, the most common is the Bayer pattern, shown in Fig. 1(a). Many algorithms for demosaicing the Bayer pattern exist in literature. In this paper we focus on low computational cost algorithm that are intended for embedded low power systems. Low cost digital cameras with CFA can acquire images at different resolutions: the most important are high resolution VGA and low resolution QVGA. In low resolution acquisition mode, the Bayer pattern is sampled to reduce the data rate. This reduction is obtained skipping two consecutive columns every four, as shown in Fig. 1(b), obtaining what we call QVGA Bayer pattern. In this paper, we propose a new solution for the interpolation of QVGA Bayer pattern, obtained sampling only columns, based on periodic reconstruction. The QVGA resolution is adopted in many devices such as cellular phones and PDAs with inexpensive cameras. The QVGA interpolation algorithm proposed in this paper could also be used to improve the color image acquisition in these handheld devices or for image visualization in small LCDs. We use QVGA low resolution mode in our Kid-Size Humanoid to reduce the data rate and to improve image processing speed, maintaining a large field of view and reducing the resolution. We use QVGA resolution for full field of view and VGA for focalized view reduced to a region of interest of the image, where an object of interest has been detected. We also introduce an evaluation function to compare the different interpolation methods for QVGA; this function shows effectiveness of improvement of quality in interpolation results due to periodic reconstruction. The remainder of the paper is organized as follows. In Section 2, we summarize the most common interpolation methods for the Bayer pattern. In Section 3, we present basic interpolation methods for QVGA Bayer pattern and
the new one proposed by us. In Section 4 these algorithms are compared using quality evaluation functions.

2 DEMOSAICING BAYER PATTERN

Consider a 2 × 2 base grid, the Bayer pattern measures the G in two diagonal pixel and the R and B in the others, as shown in Fig. 1(a). The G component is measured twice with respect to R or B because the peak sensitivity of the human visual system lies in the medium wavelengths, corresponding to the G portion of the visible spectrum. The computer vision community has devoted many efforts to the problem of Bayer pattern demosaicing. There are many algorithms to interpolate the Bayer pattern. In this paper, we consider only low computational algorithm. For a comprehensive description of the current state of the art see the work of Gunturk et al. (Gunturk et al. (2005)).

The Nearest Neighbor Replication method is the simplest algorithm for demosaicing CFA images. It is analyzed in the works of (Parulski (1985)), (Wu et al. (1997)), (Chan et al., 1996), (Tsai et al. (1997)), (Zen et al. (1998)), (Adams et al. (1998)) and also of (Sakamoto et al. (1998)). In other works, such as in (Lukin (2004a)). Bilinear Interpolation method forms the base for more sophisticated algorithm; in (Malvar (2004)) this method is studied for non–linear interpolation algorithm on rhomboidal grid. The algorithm is non–adaptive. There are two privileged directions (i.e., vertical and horizontal) and pixels are bilinearly interpolated along these directions.

An interpolation algorithm that has good performances in terms of quality of reconstruction is Linear Interpolation with Laplacian Second–order Correction Terms that we adopted in our application. The Linear Interpolation with Laplacian Second–order Correction Terms is an adaptive algorithm, i.e., it changes the interpolation equation adapting to the image to be interpolated. This algorithm is analyzed in (Hamilton and Adams (1997)) and with modifications in (Adams and Hamilton (1997)) and (Adams et al. (1997)). Interpolation of the green component has to be computed before the other components. Consider interpolation of green component in blue pixel, see Fig. 2(a), for example G5 in B5. Calculate gradient in horizontal (ΔH) and vertical (ΔV) direction

\[
\Delta H = |G4 - G6| + |B5 - B3 + B5 - B7| \quad ; \quad (1)
\]

\[
\Delta V = |G2 - G8| + |B5 - B1 + B5 - B9| \quad ; \quad (2)
\]

Then evaluate which interpolation function has to be used:

\[
\begin{align*}
\Delta H < \Delta V & \rightarrow \quad G5 = \frac{(G4 + G6) + (2 \cdot B5 - B3 - B7)}{4} \quad ; \quad (3) \\
\Delta H > \Delta V & \rightarrow \quad G5 = \frac{(G2 + G8) + (2 \cdot B5 - B1 - B9)}{4} \quad ; \quad (4) \\
\Delta H = \Delta V & \rightarrow \quad G5 = \frac{(G2 + G4 + G6 + G8) + (4 \cdot B5 - B1 - B3 - B7 - B9)}{8} \quad ; \quad (5)
\end{align*}
\]

For red and blue components are possible tree cases, consider Fig. 2(b):

1. interpolate red or blue component in green position when nearest same color components are in the same column. Consider R4 in G4, interpolation is done with (7)

\[
R4 = \frac{R1 + R7}{2} + \frac{(G4 - G1 + G4 - G7)}{4} \quad ; \quad (7)
\]

2. interpolate red or blue component in green position when nearest same color components are in the same row. Consider R2 in G2, interpolation is done with (8)

\[
R2 = \frac{R1 + R3}{2} + \frac{(G2 - G1 + G2 - G3)}{4} \quad ; \quad (8)
\]
3. interpolate red or blue component in blue or red position. Consider $R_5$ in $B_5$. Define diagonal gradient in negative direction $\Delta N$ (9) and in positive direction $\Delta P$ (10).

$$\Delta N = |R_1 - R_9| + |G_5 - G_1 + G_5 - G_9|$$ (9)

$$\Delta P = |R_3 - R_7| + |G_5 - G_3 + G_5 - G_7|$$ (10)

Then evaluate which interpolation function has to be used:

$$\Delta N < \Delta P \rightarrow R_5 = \frac{(R_1 + R_9)}{2} + \frac{(2 \cdot G_5 - G_1 - G_9)}{4}; \quad (11)$$

$$\Delta N > \Delta P \rightarrow R_5 = \frac{(R_3 + R_7)}{2} + \frac{(2 \cdot G_5 - G_3 - G_7)}{4}; \quad (12)$$

$$\Delta N = \Delta P \rightarrow R_5 = \frac{(R_1 + R_3 + R_7 + R_9)}{4} + \frac{(4 \cdot G_5 - G_1 - G_3 - G_7 - G_9)}{8}; \quad (13)$$

$$\Delta N = \Delta P \rightarrow R_5 = \frac{(R_1 + R_3 + R_7 + R_9)}{4} + \frac{(4 \cdot G_5 - G_1 - G_3 - G_7 - G_9)}{8}; \quad (14)$$

3 DEMOSAICING QVGA BAYER PATTERN

As we already said in the introduction, the QVGA Bayer pattern is obtained sampling the Bayer pattern skipping two columns every four, see Fig. 1(b). To the best of our knowledge, there is not a standard way in literature to interpolate the QVGA Bayer pattern. In this paper we propose a new interpolation method.

The main idea originates from an original work of (Tang and Lee (2004)) about interpolation of classical Bayer pattern. In that work, Tang and Lee proposed a vertex based interpolation method instead of center based interpolation method.

Similarly, we based our interpolation algorithm for QVGA on vertex interpolation mode (i.e. the interpolated pixels are centered on the vertex of the Bayer pixels). This is sketched in Fig. 3(a). The interpolated pixel is the yellow square binding 2x2 Bayer pixels. The yellow dot represents the center of the interpolated pixel. With respect to the VGA Bayer pattern, the QVGA Bayer pattern has the following peculiarities:

1. in each yellow square one color component is missing;
2. the vertical periodicity of the grid is of 2 pixels, while the horizontal is of 4 pixels;
3. in the yellow squares along the same column is missing always the same color component;

In the next paragraphs, we will show how the Nearest Neighbor Replication algorithm and the Bi-linear Interpolation algorithm could be applied to the QVGA Bayer pattern. We will see that both algorithms perform poorly, basically because they do not use a periodic pattern for the color components. We propose a new method called Periodic Reconstruction Interpolation in which some of the color components are interpolated to obtain a periodic color pattern, that is equally repeated in every yellow square of the QVGA image, as depicted in Fig. 3(b). Interpolated pixels are always painted with the corresponding color and white bar pattern.

The Nearest Neighbor Replication algorithm is the simplest non–adaptive algorithm for demosaicing QVGA images. The green component does not need to be interpolated; every yellow square contains a green pixel of the QVGA Bayer pattern. The red and the blue pixels are obtained by replicating, respectively, the nearest right pixel and the left near-
We propose an adaptive algorithm for interpolation of QVGA images, that makes an average on vertical and horizontal direction. The green component, as in Nearest Neighbor Replication algorithm, is not interpolated. For the interpolation of red and blue components, see Fig. 5, the problem arises from the non–adaptive algorithm for demosaicing QVGA images, that makes an average on vertical and horizontal direction. The green component in every yellow square is obtained as the average of the nearest green Bayer pattern. We propose first to interpolate the green components to construct the periodic grid of Fig. 3(b), from this the name Periodic Reconstruction Interpolation, and then to interpolate the red and blue components.

The green component in every yellow square is obtained as the average of the nearest green Bayer pixels weighted by their distances. As an example consider pixel $G_{23}$, as shown in Figure 6(a). The interpolation is obtained by (17)

$$G_{23} = \frac{3 \cdot G_{21} + 4 \cdot G_{12} + 4 \cdot G_{32} + 3 \cdot G_{25}}{14}. \quad (17)$$

To limit the low-pass filter effect due to the average operation, we perform it only on pixel closer than two pixels from the interpolation center.

The next step is to create a periodic structure also for these components. Consider Fig. 6(b) where green pixels have been already interpolated by the Eq. (17). We propose an adaptive algorithm for interpolation of red and blue pixels.

Consider red pixel $R_{24}$. We define diagonal gradients in negative direction as $\Delta N$ (18) and in positive direction as $\Delta P$ (19) and the horizontal gradient as $\Delta O$ (20).

$$\Delta N = |R_{22} - R_{46}| + |G_{25} - G_{3} + G_{25} - G_{47}|. \quad (18)$$

$$\Delta P = |R_{6} - R_{42}| + |G_{25} - G_{7} + G_{5} - G_{43}|. \quad (19)$$

$$\Delta O = |R_{22} - R_{26}| + |G_{25} - G_{23} + G_{25} - G_{27}|. \quad (20)$$

The value of every interpolated pixel is calculate with Algo 3.1 depending on which of the gradient is maximum. The interpolation of blue pixel is done in the same way.

Unfortunately this algorithm blurs the image by averaging over all pixels without any appropriate weighting. Even this algorithm is affected by the zipper effect.

A thorough analysis of the Nearest Neighbor Replication and Bilinear Interpolation algorithms when used for the interpolation of QVGA Bayer pattern shows that they do not correctly interpolate the green component. This problem arises from the non–periodic interpolation domain (i.e., the distance between two green pixels varies along the grid). This results in a wrong attribution to green components. We propose first to interpolate the green components to construct the periodic grid of Fig. 3(b), from this the name Periodic Reconstruction Interpolation, and then to interpolate the red and blue components.
Algorithm 3.1 QVGA Periodic Reconstruction Interpolation.

1: for all i is pixel has to be interpolated do
2: \[ A = \max(\Delta N, \Delta P, \Delta O) \]
3: if \( A = \Delta N \) then
4: \[ R24 = \frac{(R2+R46)}{2} + \frac{(2G25-G3-G47)}{4} \]
5: else if \( A = \Delta P \) then
6: \[ R24 = \frac{(R6+R42)}{2} + \frac{(2G25-G7-G43)}{4} \]
7: else if \( A = \Delta O \) then
8: \[ R24 = \frac{(R22+R26)}{2} + \frac{(2G25-G23-G27)}{4} \]
9: else
10: \[ R24 = \frac{(4R22+4R26+3R6+3R42+3R46)}{20} \]
11: end if
12: end for

4 ALGORITHM QUALITY EVALUATION

To compare the quality of the images reconstruction with different algorithms, in addition to the visual comparison of the reconstructed images, we use the dependency index developed in (Guseo T. (2006)). This index is based on the concept of entropy, see (Guseo R. (2006)).

The original and reconstructed images to be compared has to be considered as a linear array. In bivariate case the dependency index \( p_{F} \tilde{D} \) is defined as (21), where \( P \) represent the pixel’s position in the linear array and \( F \) indicates the original and reconstructed image. \( F_{H} \) and \( p_{F} \tilde{H} \) is denoted the marginal entropy respect figure (F) and position (P), with \( p_{F} I \) the joined entropy and with \( p_{F} I \) the information between original and reconstructed figure.

\[
\begin{align*}
p_{F} \tilde{D} = \frac{p_{F} I}{p_{F} H} = \frac{p_{F} H - p_{F} H}{p_{F} H}. \quad \quad (21)
\end{align*}
\]

In the trivariate case, where third qualitative variable is color (C) we consider the partial dependency index without effect of color (22)

\[
\begin{align*}
p_{F} \tilde{D}_{C} = \frac{p_{F} C H - p_{F} F C H}{p_{F} C H} = \frac{p_{F} C H - p_{F} F C H + F F C H}{p_{F} C H - C H}. \quad \quad (22)
\end{align*}
\]

The dependency indexes are for construction normalized in the range [0,1]. If two images are very similar the dependency index tends to 0.

We use dependency indexes \( p_{F} \tilde{D}_{(B)} \), \( p_{F} \tilde{D}_{(G)} \) and \( p_{F} \tilde{D}_{(T)} \) for bivariate case for each single components and \( p_{F} \tilde{D}_{(C)} \) for image in all components. We use partial dependency index without effect of color \( p_{F} \tilde{D}_{C} \) for trivariate case.

To assess to interpolation quality of different algorithm proposed, consider the Tab. 1 the which are listed dependency indexes. Periodic Reconstruction has \( p_{F} \tilde{D}_{(G)} \) index that is half than other methods. This represent a good improvement in interpolation quality. In Fig. 7 interpolation algorithms have been visually compared . It is visible the presence of zipper effect an false colors in Fig. 7(b) and Fig. 7(c) corresponding to Nearest Neighbor Replication and Bilinear Interpolation algorithim. The Fig. 7(d) obtained with Periodic Reconstruction algorithm has sharp edges and no zipper effects.

Table 1: Comparison between interpolation methods on QVGA Bayer pattern, all values are in \( 10^{-3} \).

<table>
<thead>
<tr>
<th>Dependency index</th>
<th>( p_{F} \tilde{D}_{(B)} )</th>
<th>( p_{F} \tilde{D}_{(G)} )</th>
<th>( p_{F} \tilde{D}_{(T)} )</th>
<th>( p_{F} \tilde{D}_{(C)} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>10.4765</td>
<td>5.8445</td>
<td>23.1138</td>
<td>10.2245</td>
</tr>
<tr>
<td>Bilinear</td>
<td>10.9063</td>
<td>5.8445</td>
<td>22.6590</td>
<td>10.2537</td>
</tr>
<tr>
<td>Periodic</td>
<td>7.3436</td>
<td>2.4420</td>
<td>15.2124</td>
<td>6.8667</td>
</tr>
</tbody>
</table>

Figure 7: Visive comparison between QVGA interpolation methods.

5 CONCLUSION

The use of QVGA format is important in low cost and low computational power systems as embedded PCs, PDAs and cellular phones. In this paper we present a new interpolation method for QVGA Bayer pattern: Periodic Reconstruction Interpolation. The improve-
ment of quality of the images reconstructed with proposed method is demonstrated using dependency indexes.

The platform on which this method is implemented is Robovie-M by VStone with a Renesas CPU at 40Mhz and a RAM of 256k–Word of 16–bit. The digital camera used is a CMOS camera (an OV3620 by Omnivision). Acquiring the $240 \times 240$ Bayer pattern with a QVGA resolution and interpolating it with the Periodic Reconstruction Interpolation, enables the robot to allocate memory space for three images to be used for image processing operations that cannot be done in place. We reached a performance of 2 fps on the 40Mhz CPU that has also to control 22 motors during image processing time.

At the moment of writing, we are applying the proposed method to omnidirectional images. We are evaluating an extension of Periodic Reconstruction Interpolation to other patterns, for example to pattern implemented in STMicroelectronics cameras, obtained skipping also two rows with the same ratio.

REFERENCES


