NOISE REDUCTION AND INTERLACED-TO-PROGRESSIVE CONVERSION BASED ON OPTIMAL ADAPTATION

Seungjoon Yang, Jae-Han Jung, and You-Young Jung

Digital Media R&D Center, Samsung Electronics Co., Ltd., Suwon, Korea

ABSTRACT

This paper presents an adaptation scheme for noise reduction (NR) and interlaced-to-progressive conversion (IPC) based on the optimal filter theory. The proposed adaptation scheme provides effective removal of noise and high quality conversion with considerably less flicker. NR and IPC are combined together so that NR helps the proper operation of IPC, while sharing resources and informations.

1. INTRODUCTION

As progressive scan format displays become widely available, interlaced-to-progressive conversion (IPC) becomes an important issue. Any glitches caused during the conversion can be easily seen on a large high contrast display. Current IPC methods rely on filters that adapt to skewed spectra caused by either diagonal structures or motion in images [1]. Typically, a so-called motion adaptation scheme is employed so that temporal filters are selected when tractable motion is detected, whereas spatial filters are selected otherwise. Since penalty of miss is very high, motion adaptation is biased to the usage of spatial filters. When images are corrupted with noise, detection of motion becomes more difficult and the motion adaptation rely heavily on spatial filters. Spatial filters introduce aliasing unless images contain strong directional structures. Images that contains complex structures show severe flickering due to aliasing.

This paper presents a motion adaptation scheme that is based on the optimal filter theory. Instead of relying on heuristic methods to detect motion with pixelwise field or frame differences and combinations of rank order filters, we formulate an optimization problem to select a filter adaptively. Contrast to other schemes that select several filters with several separate cost functions and mix the results based on motion detection, our approach is to find one filter with one cost function. We address two problems in formulating the optimal filter based framework: unavailable desired signals and high computational complexity. The problems are resolved by multiresolution analysis and by reducing search space, respectively. The proposed method provides considerably reduced flickering in slowly moving complex structures. We also use the same adaptation scheme in noise reduction (NR). NR and IPC are combined together so that NR helps proper operation of IPC, while sharing resources and informations. Section 2 presents the proposed adaptation scheme. Section 3 provides the NR/IPC system. Experimental results can be found in Section 4. Conclusion is drawn in Section 5.

2. OPTIMAL ADAPTATION

2.1. Optimal Filter Theory

We denote an interlaced image by $f(\mathbf{i}, k)$ and a progressive scan image by $g(\mathbf{i}, k)$, where $\mathbf{i} = (i, j)^t$ is the spatial index and k is the temporal index. For interlaced image, we use the convention that the lines are missing when $i\%2 \neq k\%2$, where $\%$ is the modulo operation.

Our goal is to find the best filter that predicts a pixel of a desired image from the given pixels of an input image. We take small neighbors of input and desired images centered at spatial location $\mathbf{i}$ and temporal location k, and denote them by $x(\mathbf{i}, k)$ and $d(\mathbf{i}, k)$, respectively. A filter $\hat{h}$ that provides the best estimate of $d(\mathbf{i}, k)$ given the input $x(\mathbf{i}, k)$ in the mean squared error (MSE) can be obtained by

$$\hat{\mathbf{h}} = \arg\min_{\mathbf{h}} E\left[(\mathbf{d} - \mathbf{h}^t\mathbf{x})^t(\mathbf{d} - \mathbf{h}^t\mathbf{x})\right] \quad (1)$$

where $\mathbf{x}$ and $\mathbf{d}$ are lexicographically ordered vectors of input and desired pixels, respectively, and $\mathbf{h}$ is a matrix that represents the operation of the filter h. We denote the above cost function by $\Phi(\mathbf{h})$. The optimal filter is given by

$$\hat{\mathbf{h}} = \mathbf{R}^{-1}\mathbf{p} \quad (2)$$

where $\mathbf{R} = E[\mathbf{x}^t\mathbf{x}]$ and $\mathbf{p} = E[\mathbf{x}^t\mathbf{d}]$ [2].

The optimal filter theory suggests how to use a filter adaptive to the input signal. If we want to use the optimal filter for every pixel, we can collect the input and desired pixels from the neighborhood, i.e. $\mathbf{x}$ and $\mathbf{d}$, and solve for the optimal filter $\hat{\mathbf{h}}$ using (2). There are two major problems in using the optimal filter as an adaptation scheme. Firstly, depending on applications, the desired output image $\mathbf{d}$ is never available. Secondly, the solution requires an inversion of a very large matrix $\mathbf{R}$. The following two subsections provide methods to go around these problems.

2.2. Multi-Resolution Analysis

Recently, similarity between images in different resolutions has been exploited for aliasing-free interpolation [3, 4]. The assumption used in these approaches is that the higher resolution image $g(i, j)$ and the lower resolution image $g(2i, 2j)$ are similar. Hence, the optimal filter is obtained by collecting the input and desired signals $\mathbf{x}$ and $\mathbf{d}$ from the lower resolution image $g(2i, 2j)$ to be used to interpolate the higher resolution image $g(i, j)$. The use of the lower resolution image $g(2i, 2j)$ instead of the higher resolution image $g(i, j)$ when solving for the optimal filter resolves the problem of unavailable desired signal.

The same idea can be extended to interlaced image sequences. There are a couple of ways to exploit similarity in interlaced image sequences. Firstly, we can assume that the higher resolution image $f(i, j, k)$ and the lower spatial resolution image $f(2i, j, k)$ or $f(2i + 1, j, k)$ are similar. Secondly, we can assume that the higher resolution image $f(i, j, k)$ and the lower temporal resolution image $f(i, j, 2k)$ or $f(i, j, 2k + 1)$ are similar. In order to exploit these similarities, we can obtain the optimal filter with $\mathbf{x}$ and $\mathbf{d}$ collected from $f(2i, j, k)$, $f(2i + 1, j, k)$, $f(i, j, 2k)$, or $f(i, j, 2k + 1)$, and apply the filter to $f(i, j, k)$.

2.3. Reduction of Search Space

The optimal filter given in (2) involves an inversion of the matrix $\mathbf{R}$. Depending on the size of the filter $\mathbf{h}$, the matrix $\mathbf{R}$ can be very large. Inverting a big matrix for each pixel to obtain the optimal filter at video rate is not an easy task.

In order to reduce the computational complexity, we restrict the space where our optimal filter lies. In the optimization problem in (1), the search space is $\mathbb{R}^p$, where p is the size of the filter. The search space is a little bit smaller with a constraint $\mathbf{h}^t\mathbf{h} = 1$. Because the search space is so huge, the solution has to be obtained either analytically or numerically. If we restrict the search space to a small set, it is possible to find the solution by an exhaustive search. In other word, we can calculate the cost $\Phi(\mathbf{h})$ for each $\mathbf{h}$ in a feasible filter set, and pick the one that achieves the minimum cost. The solution may not be optimal in $\mathbb{R}^p$, but it is still optimal given a feasible filter set.

The feasible filter set should include all the filters that are known to be effective for typical situations. We consider two types of filters. The first type of filters are ones that take the directional structures in images into account. The directional low pass filters are given by

$$h(i, j, k; s) = \begin{cases} 1/2, & \text{if } (i, j, k) = (-1, -s, 0) \text{ or } (1, s, 0) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where s defines the direction. The set of directional low pass filters is defined by

$$\mathcal{S} = \{h(i, j, k; s) | s \in [-\theta, \theta]\}. \tag{4}$$

The second type of filters are ones that take the motion in images into account. The motion compensating filters are given by

$$h(\mathbf{i}, k; \mathbf{v}) = \begin{cases} 1, & \text{if } (\mathbf{i}, k) = (\mathbf{v}, -1) \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where $\mathbf{v} = (v_i, v_j)^t$ is the motion that the filter is compensating for. The set of motion compensating filters is defined by

$$\mathcal{T} = \{h(\mathbf{i}, k; \mathbf{v}) | v_i \in [-sr_i, sr_i], v_j \in [-sr_j, sr_j]\}. \tag{6}$$

With the two feasible filter set, we solve for the optimization problem

$$\hat{\mathbf{h}} = \arg \min_{\mathbf{h} \in \mathcal{S} \cup \mathcal{T}} E\left[(\mathbf{d} - \mathbf{h}^t\mathbf{x})^t(\mathbf{d} - \mathbf{h}^t\mathbf{x})\right]. \tag{7}$$

The size of the search space is reduced from $\mathbb{R}^p$ to $\|\mathcal{S}\| + \|\mathcal{T}\|$. It is now possible to calculate the cost for every filter in the sets, and pick one with the minimum cost.

2.4. Discussions

Because we restrict the filter to be in a very limited set, the optimality in MSE sense may be in question. The optimization problem selects the filter from the feasible set that provides the smallest MSE given the pixels in the neighborhood. Because the filter is selected adaptively given the neighborhood pixels, we call the proposed method optimal adaptation instead of optimal filtering.

The proposed method includes the directional filters and the motion compensating filters in the feasible image sets. Compared to other methods that find a couple of filters separately based on two different cost functions and mix the results, our method finds one filter based on one cost function. That is possible because we put the adaptation scheme as a MSE optimization framework.

In [5], the idea of using the optimal filter in IPC and scaling applications has been exploited. In their approach, the problem of unavailable desired data is resolved by working with a training set of low resolution-high resolution image pairs. The problem of the matrix inversion is resolved by first preparing a finite number of filters trained off-line with a low resolution-high resolution image pair training set, and by picking a filter based on classification of the input low resolution image. Compared to this approach, our idea is to utilize the similarity that can be extracted from the input image itself, and to pick a filter based on the actual MSE instead of based on classification. In [6], the optimal L^l filter is considered for IPC. In their approach, the filter is restricted to a type of rank order filter, which is trained off-line with a reference image sequence.

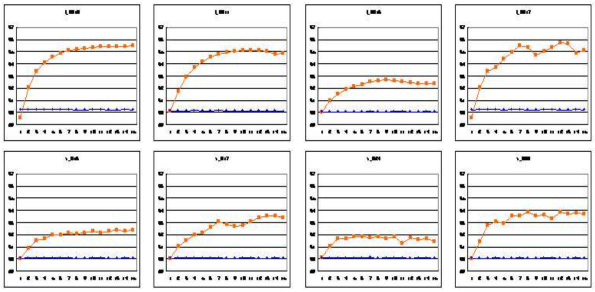

Fig. 1. PSNR improvement by NR, PSNR in dB vs. framenumber, orange line: noise reduced images, blue line: noisy images, each vertical grid corresponds to 1dB.

3. NR/IPC SYSTEM

3.1. Overview

The proposed NR/IPC system works with four consecutive field images, $\hat{f}(k+1)$, $\hat{f}(k)$, $\hat{f}(k-1)$ and $f(k-2)$, where $\hat{f}$ is the noise removed image and f is the noisy image. Combining NR and IPC together has several advantages. The overall system requirement is reduced by sharing data bandwidth and line memory. Information extracted during NR can be used in IPC. NR proceeds IPC so that the removal of noise helps the proper operation of the IPC.

3.2. Noise Reduction

NR works with the two field images of equal parities. We try to find the optimal filter that predicts the given pixels in $f(\mathbf{i}, k-2)$ from the given pixels in $\hat{f}(\mathbf{i}, k)$. The feasible filter set is $\mathcal{T}$ given in (6). Since $f(\mathbf{i}, k) = 0$ for $i\%2 \neq k\%2$, we use only the vectors $\mathbf{v}$ with even vertical components. We use the neighbors of $[5, 17]$ to calculate the expectation, which contains three given lines. The filter $\hat{h}(\mathbf{i})$ that achieves the minimum is selected, and also the minimum cost $\Phi(\hat{h}(\mathbf{i}))$ is noted. Since the cost is the mean squared prediction error by the chosen filter, it can be used to determine how accurate the prediction is. The noise reduced image is give by

$$\hat{f}(\mathbf{i}, k-2) = w \cdot f(\mathbf{i}, k-2) + (1-w) \cdot \hat{h}(\mathbf{i}) * \hat{f}(\mathbf{i}, k) \quad (8)$$

where the weight w is the normalized minimum cost $\Phi(\hat{h}(\mathbf{i}))$.

3.3. Interlaced-to-Progressive Conversion

IPC works with three field images. We try to find the optimal filter that predicts the given pixels in $\hat{f}(\mathbf{i}, k-1)$ from the given pixels in $\hat{f}(\mathbf{i}, k+1)$. We use a feasible filter set that is smaller than $\mathcal{T}$. The feasible filter set, denoted by $\mathcal{C}$, is the filters selected by the NR for the four causal neighbors of $\mathbf{i}$ and the filter that compensates for zero motion, $h(\mathbf{i}, k; \mathbf{0})$. We also try to find the optimal filter that predicts the given pixels in $\hat{f}(\mathbf{i}, k)$ from the given pixels in $\hat{f}(\mathbf{i}, k)$. The feasible filter set is $\mathcal{S}$ in (4). We use the neighbor of $[3, 17]$ to calculate the expectation, which contains two given lines. Since the set $\mathcal{C}$ and $\mathcal{S}$ are disjoint, we have

$$\min_{h \in \mathcal{C} \cup \mathcal{S}} \Phi(h) = \min\{\min_{h \in \mathcal{C}} \Phi(h), \min_{h \in \mathcal{S}} \Phi(h)\}. \quad (9)$$

Hence, we select a filter from each feasible set, and find the one that achieves lower cost. The chosen filter is used to interpolate the missing pixel in $\hat{f}(\mathbf{i}, k)$ to obtain the progressive scan format image $g(\mathbf{i}, k)$. Since we find the optimal filter at lower resolution, the filter has to be adjusted to higher resolution by a factor of two; $h(\mathbf{i}, k; s/2)$ is used instead of $h(\mathbf{i}, k; s)$, or $h(\mathbf{i}, k; \mathbf{v}/2)$ is used instead of $h(\mathbf{i}, k; \mathbf{v})$.

The above adaptation scheme works well with most of images. However, when images contains unusually high vertical frequency, for example lines of single pixel height, or unusually high temporal frequency, for example very fast motion, the proposed scheme malfunctions. To cope with these situations, we implement counters that register history of which filters are chosen, and based on the counter values we overwrite the selection [7].

(a)

(b)

Fig. 2. An example of NR, a frame from captured broadcasting images, (a) noisy image, (b) noise removed image.

(a)

(b)

Fig. 3. An example of IPC, a frame from a test DVD, (a) by a motion adaptive IPC, (b) by the proposed method.

4. EXPERIMENTS AND DISCUSSIONS

Currently, the proposed NR/IPC system is implemented in c++. We set $\theta = 6$, $sr_i = 4$, and $sr_j = 8$. As a result we are considering 13 filters in $\mathcal{S}$, 85 filters in $\mathcal{T}$, and 5 filters (selected adpatively from $\mathcal{T}$) in $\mathcal{C}$. In order to maintain reasonable complexity, we find a filter for every $[1, 8]$ segment for the filters in $\mathcal{T}$ and $\mathcal{C}$, but update the cost $\Phi(h)$ for every pixels. Experiments are performed with images in our vast image database, which consists of several test DVD's, captured broadcasting signals, etc.

Fig. 1 shows peak signal to noise ratio (PSNR) improvement after NR. Zero mean Gaussian noise with varaince $\sigma = 8$ is added to the original interlaced images. Typically, PSNR improvements of 2dB to 5dB are reported depending on image contents; lower values for images with complex motion and higher values for images with simple motion. Fig. 2 shows an example of NR with actual noisy broadcasting images. Since we use motion compensating filters, NR results are quite effective even near moving objects, for example near the model's moving arm.

The performance of IPC is hard to evaluate because original progressive images are usually not available. The goal of our method is to achieve proper adaptation in slowly moving area so that annoying flickering can be reduced. Since flickering is a characteristics of motion pictures, it is even harder to demonstrate it in a paper with an image. Fig. 3 shows an example of IPC results. Images of a slowly moving book are used for this example. The image in (a) is the result of a typical motion adaptive IPC. When motion adaptation scheme detects the motion, it puts heavy weights on spatial interpolation that limits the vertical resolution. The limitation of resolution that happens in every frame in different places in different ways leads to severe flickering. The proposed method selects motion compensating filters that provide higher vertical resolution. Since the resolution is maintained throughout the frames, there is less flickering. The proposed method also selects directional filters so that the round frames of the eye glasses are properly converted without aliasing.

5. CONCLUSION

This paper provides a method to select filters that adapt to skewed spectra due to structures and motion based on the optimal filter theory. The proposed method is used in NR and IPC, which are combined together for effective and efficient operations.

6. REFERENCES

[1] E.B. Bellers and G. de Haan, *De-interlacing: A Key Technology for Scan Rate Conversion*, Elsevier, Amsterdam, 2000.

[2] S. Haykin, *Adaptive Filter Theory*, Prentice hall, New Jersey, 2nd edition, 1991.

[3] X. Li and M.T. Orchard, "New edge-directed interpolation", *IEEE Trans. Image Processing*, vol. 10, pp. 1521–1527, Oct. 2001.

[4] D.D. Muresan and T.W. Parks, "Optimal recovery approach to image interpolation", *IEEE Int. Conf. Image Processing*, vol. 3, pp. 848–851, Oct. 2001.

[5] T. Kondo, Y. Fujimori, H. Nakaya, and K. Takahashi, "Signal converting apparatus and signal converting method", *United States Patent 5,852,470*, Dec. 1998.

[6] J. Schwendowius and G.R. Arce, "Data-adaptive digital video format conversion algorithms", *IEEE Trans. Circuits and Syst. for Video Technol.*, , no. 3, pp. 511–526, June 1997.

[7] S. Yang, Y.Y. Jung, Y.H. Lee, and R.-H. Park, "Motion compensation assisted motion adaptive interlaced-to-progressive conversion", *IEEE Int. Conf. Image Processing*, vol. 2, pp. 909–912, Sept. 2003.