Fast template matching in images under non-linear tone mapping

Yacov Hel-Or Hagit Hel-Or Eyal David

Technical Report
Efi Arazi School of Computer Science
The Interdisciplinary Center, Herzliya, Israel,
March 2012

Abstract

A fast pattern matching scheme termed Matching by Tone Mapping (MTM) is introduced which allows matching under non-linear tone mappings. We show that, when tone mapping is approximated by a piecewise constant/linear function, a fast computational scheme is possible requiring computational time similar to the fast implementation of Normalized Cross Correlation (NCC). In fact, the MTM measure can be viewed as a generalization of the NCC for non-linear mappings and it actually reduces to NCC when mappings are restricted to be linear. We empirically show that the MTM is highly discriminative and robust to noise with comparable performance capability to that of the well performing Mutual Information, but on par with NCC in terms of computation time.

Keywords: Pattern matching, template matching, structural similarity, photometric invariance, Matching by Tone Mapping, MTM, nonlinear tone mapping.

1 Introduction

Template or pattern matching is a basic and fundamental image operation. In its simple form a given pattern is sought in an image, typically by scanning the image and evaluating a similarity measure between the pattern and every image window. Fast and reliable pattern matching is a basic building block in a vast range of applications, such as: image denoising, image re-targeting and summarization, image editing, super-resolution, object tracking and object recognition, and more (e.g. [7, 32, 19, 2]).

In most cases, however, the input image is acquired in an uncontrolled environment, thus, the sought pattern may vary in tone-levels due to changes in illumination conditions, camera photometric parameters, viewing positions, different modalities, etc. [21]. Commonly, these changes can be modeled locally by a non-linear tone mapping - a functional mapping between the gray-levels of the sought pattern and those of the image pattern. In this paper we deal with pattern matching where gray-levels may be subject to some unknown, possibly non-linear, tone mapping.

When dealing with matching under tone-mapping, three classes of approaches have been considered: The first approach attempts to determine local signatures within the pattern and image that are invariant to tone mapping. Examples of this approach include Gradient Signatures [16], Histogram of Gradients [12], SIFT [22] and others (see [23] for comparative study). These signatures are often encoded to be invariant to geometric transformations as well as tone variations. However, the data contraction implemented by these methods inherently involve loss of information and, thus, weaken their discrimination power. Consequently, these techniques often requiring an addition verification phase. Another approach to matching under tone-mapping, involves transformation of the pattern and image into a canonical configuration. Examples of this approach include Histogram Equalization and the well known Normalized Cross-Correlation [6]. These approaches are limited in that there is no known canonical configuration for non-linear mappings. Finally, brute force methods attempt to perform template matching by searching the entire transformation parameter space, resulting in highly time consuming methods. Many distance measures for pattern matching have been suggested in the literature and the interested reader is referred to [11, 6] for excellent reviews. The approach suggested in this paper involves a search in the tone-mapping parameter space, however this search is performed very efficiently in closed form.

By far, the most common distance measure used in template matching is the Euclidean distance. Assume the pattern $\mathbf{p}$ and the candidate window $\mathbf{w}$ are both vectors in $\mathcal{R}^m$, (e.g. by raster scanning the pixels). The Euclidean distance between $\mathbf{p}$ and $\mathbf{w}$ is denoted: $d_E(\mathbf{p},\mathbf{q}) = \|\mathbf{p}-\mathbf{w}\|_2$. Searching for the pattern over the entire im-

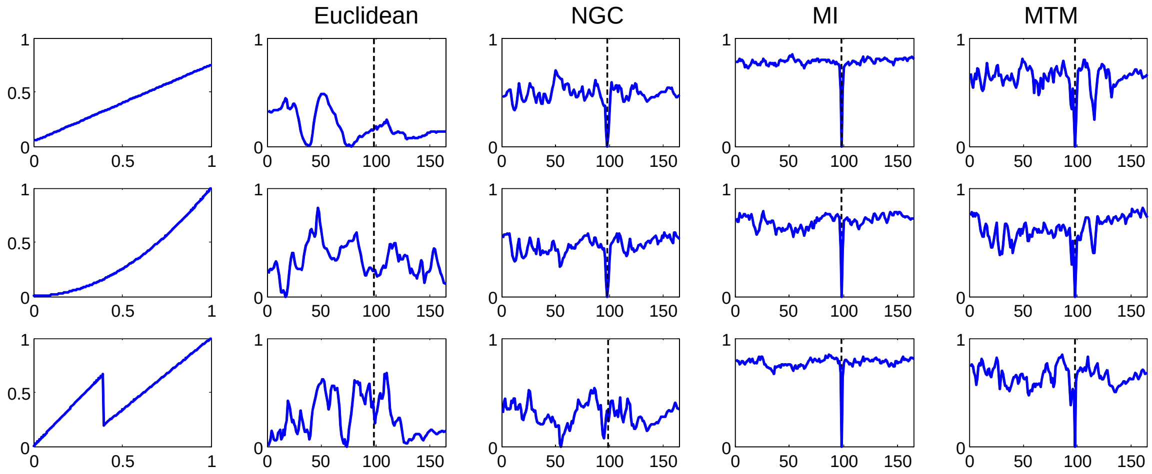

Figure 1: Pattern Matching under tone mapping. Euclidean, NCC, MI and MTM distance measures were used to evaluate distances between a pattern and all image windows for different tone mappings. To simplify visualization results are shown as the minimum distance value between pattern and image windows along image columns. Distance values have been normalized to the range $[0, 1]$ for easier comparison. Lower values indicate greater similarity of the window to the pattern. The dashed line marks the column of the correct match. Rows (top to bottom): Linear, Monotonic and Non-Monotonic tone mappings. Columns (left to right): Tone mapping function, followed by plots for Euclidean, NCC, MI (using bins of 20 tone-values) and MTM (using bins of 20 tone-values).

age is performed by scanning the image and determining the minimal d_E value. This search can be applied very fast using efficient convolution schemes [14]. Nevertheless, although very common, the Euclidean distance assumes that no tone mapping has been applied or, equivalently, that the tone mapping between $\mathbf{p}$ and $\mathbf{w}$ is the identity mapping. Clearly, the Euclidean distance is inadequate when the image undergoes tone deformations as demonstrated in Figure 1.

To overcome gray tone mapping effects in images, the *normalized cross correlation* (NCC) distance is often used [6]. Consider the pattern $\mathbf{p}$ and the candidate window $\mathbf{w}$ as random variables with samples p_i and w_i, $i = 1..m$, respectively. The NCC is then defined as:

$$\rho(\mathbf{p}, \mathbf{w}) = E\left[\left(\frac{\mathbf{p} - E[\mathbf{p}]}{\sqrt{\mathrm{var}(\mathbf{p})}}\right)\left(\frac{\mathbf{w} - E[\mathbf{w}]}{\sqrt{\mathrm{var}(\mathbf{w})}}\right)\right]$$

where for any vectors $\mathbf{x} \in \mathcal{R}^m$, $E[\mathbf{x}]$ and $\mathrm{var}(\mathbf{x})$ denote the sample mean and variance. Due to the substraction of the mean and normalization by the s.t.d. in both $\mathbf{p}$ and $\mathbf{w}$, the NCC distance is invariant to linear tone mappings (Figure 1 - top). The NCC distance can be applied very efficiently requiring little more than a single convolution on the input image [20]. However, such a distance will fail to detect patterns in cases where non-linear tone mappings have been applied (Figure 1 bottom row)[1].

In many cases, the tone mapping is non-linear but still maintains monotonicity, namely, the order of tone-levels is preserved under the mapping. This scenario is common between images acquired using different cameras whose internal photometric parameters differ (tone correction, sensor spectral sensitivity, white balancing, etc.). An example of such a mapping is presented in Figure 1 (middle row). Distance measures that are based on ordinal values rather than the tone-levels themselves can account for monotonic mappings. Examples of such measures, include *Local Binary Pattern (LBP)* [24], *Speeded Up Robust Features (SURF)* [3] and *Binary Robust Independent Elementary Features (BRIEF)* [9]. The LBP is an image representation where each pixel is assigned a rank value according to its ordinal relationships with its immediate neighboring pixels. The SURF and BRIEF methods encode the ordinal relationship between a sequence of (random) pairs of pixels within a pattern as a binary string. These representations are invariant to monotonic tone mapping and thus can be used to detect patterns in such cases. These approaches are fast to apply but, as will be shown below, they are very sensitive to noise. Furthermore, these measures fail under non-monotonic mappings.

Finally, when non-linear mapping is considered (Figure 1 - bottom row), the *Mutual Information* (MI) is commonly used, initially proposed for image registration [35]. MI measures the statistical dependency between two variables. Clearly, the statistical dependency is strong when gray-levels of one image result from a functional mapping of the gray-levels of the other image. Thus, MI can account for non-linear mappings (both monotonic and non-monotonic as in Figure 1).

In the context of pattern matching, MI measures the loss of entropy in the pattern $\mathbf{p}$ given a candidate window $\mathbf{w}$:

$$MI(\mathbf{p},\mathbf{w}) = H(\mathbf{w}) - H(\mathbf{w}|\mathbf{p}) = H(\mathbf{w}) + H(\mathbf{p}) - H(\mathbf{w},\mathbf{p})$$

where H is the differential entropy.

Although MI is an effective distance measure that can account for non-linear mappings, it is hindered by computational issues. First, it is computationally expensive as it requires the construction of the joint histogram (pattern vs. window) for each window to be matched. Although fast methods for evaluating histograms on running windows have been suggested [27, 36], fast methods for calculating local *joint* histograms are

---

[1] NCC often performs well even under monotonic non-linear mappings as these can often be assumed to be locally linear.

yet a challenge. Additionally, entropy as well as MI is very sensitive to the the size of histogram bins used to estimate the joint density, especially when sparse samples are given (small pattern size). Using kernel density estimation methods [34] rather than discrete histograms is, again, computationally expensive when dealing with joint probability, not to mention its sensitivity to the kernel width.

In this paper we propose a very fast pattern matching scheme based on a distance measure expressed as a minimization problem over all possible tone mappings, thus, we term the resulting measure *Matching by Tone Mapping (MTM)*. From its definition, the MTM is invariant to non-linear tone mappings and can be viewed as a generalization of the NCC for non-linear mappings and actually reduces to NCC when mappings are restricted to be linear [30]. In fact, for the general case, it can be shown that the MTM measure coincides with the statistical measure known as the correlation-ratio [15]. This measure was previously suggested as an image similarity measure in the context of multi-modal image registration [29, 31]. In this paper we show how this measure can be adapted to pattern matching, and suggest a very fast computational scheme requiring computational time similar to the fast implementation of NCC. Additionally, we extend and modified the statistical scheme to be appropriate in cases of small samples, as typically occurs when matching is applied on small patches. We empirically show that the MTM is highly discriminative and robust to noise with comparable performance capability to that of the well performing Mutual Information. Thus, the MTM allows a pattern matching scheme on par with NCC in terms of computation time but with performance capability equivalent to that of the Mutual Information scheme.

2 Matching by Tone Mapping

In the proposed pattern matching scheme, we wish to evaluate the minimum distance between a pattern and a candidate window under all possible tone mappings. Since tone mapping is not necessarily a bijective mapping, two alternatives may be considered: i) tone mapping applied to the pattern, transforming it to be as similar as possible to the candidate window, and ii) tone mapping applied to the window, transforming it to be as similar as possible to the pattern. For each case we find the minimum normed distance over all possible tone mappings.

Let $\mathbf{p} \in \mathcal{R}^m$ be a pattern and $\mathbf{w} \in \mathcal{R}^m$ a candidate window to be compared against. Denote by $\mathcal{M} : \mathcal{R} \rightarrow \mathcal{R}$ a tone mapping function. Thus, $\mathcal{M}(\mathbf{p})$ represents the tone mapping applied independently to each entry in $\mathbf{p}$. For the case of tone mapping applied to the pattern, the MTM distance is defined as follows:

$$D(\mathbf{p}, \mathbf{w}) = \min_{\mathcal{M}} \left\{ \frac{\| \mathcal{M}(\mathbf{p}) - \mathbf{w} \|^2}{m \ \mathrm{var}(\mathbf{w})} \right\} \tag{1}$$

Similarly, if the mapping is applied to the window rather than the pattern, we define:

$$D(\mathbf{w}, \mathbf{p}) = \min_{\mathcal{M}} \left\{ \frac{\| \mathcal{M}(\mathbf{w}) - \mathbf{p} \|^2}{m \ \mathrm{var}(\mathbf{p})} \right\} \tag{2}$$

The numerator in both cases is simply the norm distance after compensating for the tone mapping. The denominator is a normalization factor enforcing the distance to be scale invariant. Thus $D(\mathbf{p}, \mathbf{w}) = D(\mathbf{p}, \alpha\mathbf{w})$ for any scalar α. Additionally, it penalizes incorrect matching of $\mathbf{p}$ to smooth windows when using the constant mapping $\mathcal{M}(\mathbf{p}) = c$. Due to the tone mapping compensation, the MTM measure reflects the inherent *structural* similarity between the pattern $\mathbf{p}$ and the window $\mathbf{w}$.

Searching for the pattern in the entire input image requires calculating the optimal tone mapping for each possible window in the image. Although seemingly a computationally expensive process, we show in the following sections that in fact this distance can be calculated very efficiently requiring an order of a single convolution with the input image!

In the next section we introduce the *Slice Transform* [18]. This transform enables the representation of tone mappings in a linear form allowing a closed form solution for the defined MTM distance.

2.1 The Slice Transform (SLT)

The *Slice Transform (SLT)* was first introduced in [18] in the context of Image Denoising. In this paper we exploit the SLT to represent a mapping function using a linear sum of basis functions. We first introduce a simplified version of the transform: the Piecewise Constant (PWC) case. Consider an image segment represented as a column vector $\mathbf{x} = [x_1, \cdots, x_m]^T$ with values in the half open interval $[a, b)$. The interval is divided into k bins with boundary values $q_1 \cdots q_{k+1}$ such that:

$$a = q_1 < q_2 < \ldots < \quad q_{k+1} = b$$

Any value $v \in [a, b)$ is naturally associated with a single bin $\pi(v) \in \{1 \cdots k\}$:

$$\pi(v) = i \quad \text{if} \quad v \in [q_i, q_{i+1})$$

Given the bins defined by $\{q_i\}$, the vector $\mathbf{x}$ can be decomposed into a collection of binary *slices*: Slice $\mathbf{x}^i = [x^i_1, \cdots, x^i_m]$ is an indicator function over the vector $\mathbf{x}$

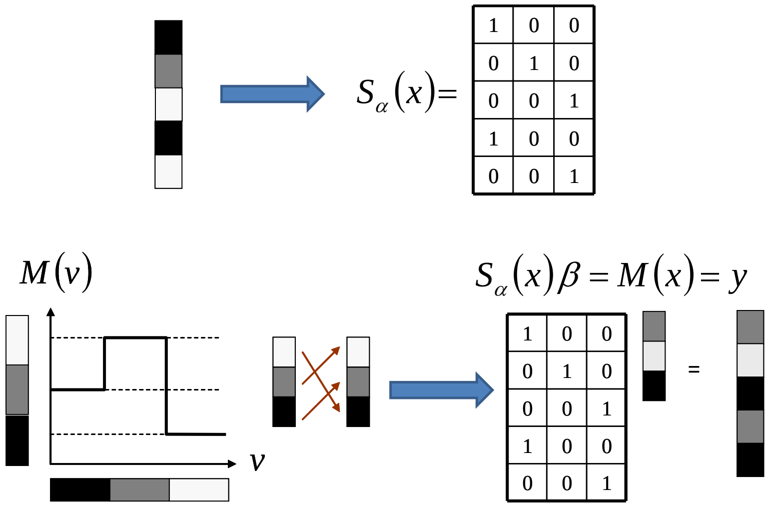

Figure 2: Top: the SLT matrix for a 5-pixel vector having 3 gray values. Bottom: a piecewise constant mapping and its representation using the SLT matrix.

representing the entries of $\mathbf{x}$ associated with the i-th bin.

$$x_j^i = \begin{cases} 1 & \text{if } \pi(x_j) = i \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The vector $\mathbf{x}$ can then be approximated as a linear combination of slice images:

$$\mathbf{x} \approx \sum_{i=1}^{k} \alpha_i \mathbf{x}^i \tag{4}$$

where the weights $\{\alpha_i\}_{i=1}^k$ are the values assigned for each bin (e.g $\alpha_i = q_i$ or $\alpha_i = (q_i + q_{i+1})/2$). The approximation is in fact a quantization of the values of $\mathbf{x}$ into the bins represented by $\{\alpha_i\}$. The greater the number of bins the better the approximation of the original image. In particular, if $\mathbf{x}$ values are discrete and $\forall j \; x_j \in \{q_i\}_{i=1}^{k+1}$ then $\mathbf{x} = \sum_{i=1}^k \alpha_i \mathbf{x}^i$.

Collecting the slices $\mathbf{x}^i$ in columns, we define the *SLT matrix* of $\mathbf{x}$:

$$S(\mathbf{x}) = [\mathbf{x}^1, \mathbf{x}^2, \cdots, \mathbf{x}^k] \tag{5}$$

Then Equation 4 can be rewritten in matrix form:

$$\mathbf{x} \approx S(\mathbf{x})\alpha \tag{6}$$

where we define $\alpha = [\alpha_1, \alpha_2, \cdots, \alpha_k]^T$. Note, that since the slices are mutually disjoint, the columns of $S(\mathbf{x})$ are mutually orthogonal, satisfying:

$$\mathbf{x}^i \cdot \mathbf{x}^j = |\mathbf{x}^i| \; \delta_{i,j} \tag{7}$$

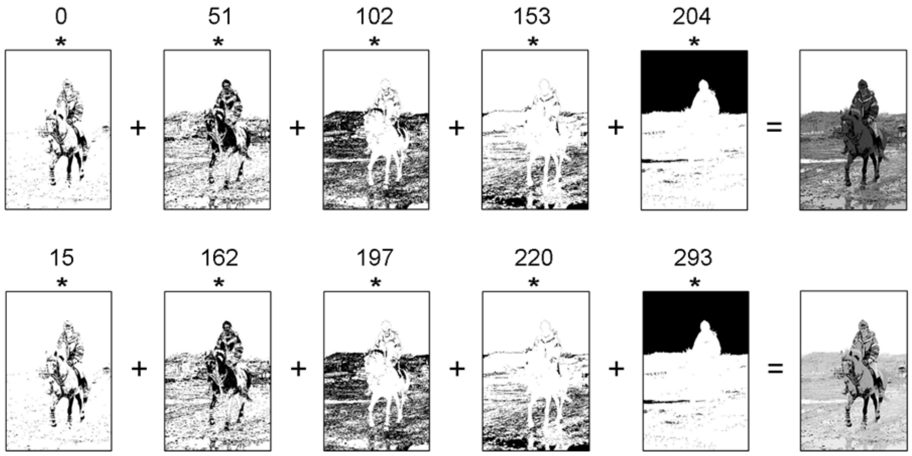

Figure 3: Linear Combination of image slices. The SLT transform was applied to an image using 5 bins defined by $\alpha = [0, 51, 102, 153, 204, 256]$. Top: Reconstructing the original images using α values as weights in the linear combination. Bottom: Using weights other than α produces a tone mapped version of the original image. Slice images are shown inverted (1=black, 0=white).

where $'\cdot'$ is the vectorial inner product, $|\mathbf{x}|$ denotes the cardinality of $\mathbf{x}$ and $\delta_{i,j}$ is the Kronecker's delta. The SLT matrix enables the representation of any piece-wise constant mapping of $\mathbf{x}$. In fact, substituting the vector α in Equation 6 with a different vector β, we obtain

$$\mathbf{y} = S(\mathbf{x})\beta \tag{8}$$

Image $\mathbf{y}$ is a piecewise constant tone mapping of $\mathbf{x}$ s.t. all pixels of $\mathbf{x}$ with values in the j-th bin are mapped to β_j. Thus, the columns of $S(\mathbf{x})$ form an orthogonal basis spanning the space of all images that can be produced by applying a piecewise constant tone mapping on $\mathbf{x}$. Figure 2 illustrates an SLT matrix (top) and a piecewise mapping of a 5-pixel signal with 3 gray-level values (bottom). Figure 3 shows an example of linearly combining image slices to form the original (quantized) image (top row) and to form a tone mapped version (bottom row).

In the context of this paper, we use the SLT for tone mapping approximation. A mapping applied to pattern $\mathbf{p}$ is approximated by a piecewise constant mapping:

$$\mathcal{M}(\mathbf{p}) \approx S(\mathbf{p})\beta$$

Consequently, the distance measures as defined in Equations 1-2 can be rewritten using the SLT:

$$D(\mathbf{p}, \mathbf{w}) = \min_{\beta} \frac{\parallel S(\mathbf{p})\beta - \mathbf{w} \parallel^2}{m \ \mathrm{var}(\mathbf{w})} \tag{9}$$

and similarly

$$D(\mathbf{w},\mathbf{p}) = \min_{\beta} \frac{\parallel S(\mathbf{w})\beta - \mathbf{p} \parallel^2}{m \ \mathrm{var}(\mathbf{p})} \tag{10}$$

where $S(\mathbf{p})$ and $S(\mathbf{w})$ are the SLT matrices as defined in Equation 5. In the following sections we elaborate on the two possible alternatives defined above. We show that solving for D for each image window can be applied in a very efficient manner. In fact, computing D for the entire image requires on the order of a single image convolution.

2.2 MTM Distance Measure using SLT

The SLT scheme allows a closed form solution for the minimizations defined in Equations 1 and 2. To introduce the matching process, we first consider the pattern-to-window case where a pattern $\mathbf{p}$ is to be matched against a candidate window $\mathbf{w}$. Thus, the distance measure used is that given in Equation 9. To simplify notation, we henceforth denote the SLT matrix $S(\mathbf{p})$ as S. The solution for β that minimizes Equation 9 is given by:

$$\hat{\beta} = \arg\min_{\beta} \|S\beta - \mathbf{w}\|^2 = S^{\dagger} \ \mathbf{w}$$

where $S^{\dagger} = (S^T S)^{-1} S^T$ is the Moore-Penrose pseudo-inverse. Substituting into Equation 9 we obtain:

$$D(\mathbf{p},\mathbf{w}) = \frac{\parallel S\hat{\beta} - \mathbf{w}\|^2}{m \ \mathrm{var}(\mathbf{w})} = \frac{\|S(S^T S)^{-1} S^T \mathbf{w} - \mathbf{w}\|^2}{m \ \mathrm{var}(\mathbf{w})}$$

Due to the orthogonality of S, we have that $G = S^T S$ is a diagonal matrix with the histogram of $\mathbf{p}$ along its diagonal: $G(i,i) = |\mathbf{p}^i|$ where $\mathbf{p}^i$ is the pattern slice associated with the i-th bin as defined in Equation 3. Expanding the numerator it is easy to verify that:

$$\|S(S^T S)^{-1} S^T \mathbf{w} - \mathbf{w}\|^2 = \|\mathbf{w}\|^2 - \|G^{-1/2} S^T \mathbf{w}\|^2$$

Exploiting the diagonality of G and using $S = \left[\mathbf{p}^1, \mathbf{p}^2, \cdots, \mathbf{p}^k\right]$, the above expression can be re-written using a sum of inner-products:

$$\|\mathbf{w}\|^2 - \|G^{-1/2} S^T \mathbf{w}\|^2 = \|\mathbf{w}\|^2 - \sum_j \frac{1}{|\mathbf{p}^j|} (\mathbf{p}^j \cdot \mathbf{w})^2$$

As a result, the overall MTM distance $D(\mathbf{p},\mathbf{w})$ reads:

$$D(\mathbf{p},\mathbf{w}) = \frac{1}{m \ \mathrm{var}(\mathbf{w})} \left[\|\mathbf{w}\|^2 - \sum_j \frac{1}{|\mathbf{p}^j|} (\mathbf{p}^j \cdot \mathbf{w})^2 \right] \tag{11}$$

In a similar manner, when matching is applied by mapping $\mathbf{w}$ towards $\mathbf{p}$ (window-to-pattern), we use Equation 10 and exchange the role of $\mathbf{w}$ and $\mathbf{p}$ to obtain a symmetric expression:

$$D(\mathbf{w},\mathbf{p}) = \frac{1}{m\ \mathrm{var}(\mathbf{p})}\left[\|\mathbf{p}\|^2 - \sum_j \frac{1}{|\mathbf{w}^j|}(\mathbf{w}^j \cdot \mathbf{p})^2\right] \tag{12}$$

2.3 Calculating MTM Distance Over an Image

Equations 11 and 12 provide a method for computing the structural difference between $\mathbf{p}$ and $\mathbf{w}$ using two complementary distances. For pattern matching, this computation must be performed on each candidate window of a given input image. Naively applying the above expressions to each image window is highly time consuming and impractical. In the following we show that, in fact, computing $D(\mathbf{p},\mathbf{w})$ or $D(\mathbf{w},\mathbf{p})$ over an entire image can be calculated very efficiently. We first describe the pattern-to-window mapping case, and then detail the window-to-pattern case.

P2W - Mapping pattern to window

Let $\mathbf{F}$ be a 2D image with n pixels in which the pattern $\mathbf{p}$ is sought. Denote by $\mathbf{w}_r$ the r-th window of $\mathbf{F}$, $r \in \{1, \cdots, n\}$. Consider the pattern-to-window (P2W) scheme where the distance given in Equation 11 is used. For each window $\mathbf{w}_r \in F$ two terms must be calculated, namely the numerator and the denominator of Equation 11:

$$d_1(r) = \|\mathbf{w}_r\|^2 - \sum_j \frac{1}{|\mathbf{p}^j|}(\mathbf{p}^j \cdot \mathbf{w}_r)^2 \quad \text{and} \quad d_2(r) = m\ \mathrm{var}(\mathbf{w}_r)$$

Since computing the inner-product over all windows can be performed using image convolution, the terms above can be calculated efficiently. We use $\mathrm{var}(\mathbf{w}_r) = E\left[\mathbf{w}_r^2\right] - E^2\left[\mathbf{w}_r\right]$ to efficiently calculate the denominator. Algorithm 1 gives the pseudo-code for calculating the P2W MTM distance between pattern $\mathbf{p}$ and each window in F (code can be found in [17]). In the pseudo-code '$*$' denotes image convolution[2], $\odot$ and $\oslash$ denote elementwise multiplication and division, respectively. We denote by upper-case letters arrays of size similar to that of the image F, and by lower-case letters scalar variables. Vectors and filter kernels are denoted by bold lower-case letters. $\mathbf{1}$ is an m-vector of 1's (box filter).

Prior to the loop, two convolutions with a box filter are calculated, each of which can be applied efficiently (with a separable 1D box filter) requiring a total of 4 additions per pixel. Within the loop there are k convolutions with the pattern slices $\{\mathbf{p}^j\}_{j=1}^k$.

---

[2] In fact, correlations rather than convolutions are required, thus, a flipped kernel is used when needed.

Algorithm 1 MTM - Pattern-to-Window

```
{Input: pattern $\mathbf{p}$, image $F$.}
{Output: image $D$ of MTM distances.}
$W_1 := \mathbf{1} * F$    {window's sum}
$W_2 := \mathbf{1} * (F \odot F)$    {window's sum of squares}
$D_2 := W_2 - (W_1 \odot W_1)/m$    {calc $d_2$ (denominator)}
Generate $\{\mathbf{p}^j\}$, for $j = 1..k$
$D_1 := 0$    {will accumulate the numerator}
for $j := 1$ to $k$ do
    $n = \mathbf{1} \cdot \mathbf{p}^j$,    {calc $|\mathbf{p}^j|$}
    $T := flip(\mathbf{p}^j) * F$    {convolve image with slice $j$}
    $T := (T \odot T)/n$
    $D_1 := D_1 + T$
end for
$D := (W_2 - D_1) \oslash D_2$
return D
```

Since each slice $\mathbf{p}^j$ is sparse, convolving it with an image requires only $|\mathbf{p}^j|$ additions per pixel using a sparse convolution scheme [37]. Additionally, since all pattern slices are mutually disjoint the number of additions sum up to a total of m additions per pixel. All other operations sum to $O(k)$ operations per pixel, thus, the algorithm requires a total of $O(mn + kn)$ operations which is comparable in complexity to a single convolution! Memory requirement is also economized. Distance value for each image window is accumulated in place, thus the required memory is on the order of the image size.

W2P - Mapping window to pattern

Consider now the window-to-pattern (W2P) scheme using the distance given in Equation 12. For each window $\mathbf{w}_r \in F$, the expressions to be calculated are:

$$d_1(r) = \|\mathbf{p}\|^2 - \sum_j \frac{1}{|\mathbf{w}_r^j|}(\mathbf{w}_r^j \cdot \mathbf{p})^2 \quad \text{and} \quad d_2 = m \ \mathrm{var}(\mathbf{p})$$

d_2 and the first term of d_1 are constant for all windows and are calculated only once. The second term in d_1 differs for each window. Algorithm 2 gives the pseudo-code for calculating the W2P distance over the entire image. In this algorithm F^j denotes the j-th image slice, i.e. $F^j(x, y) = 1$ iff $\pi(F(x, y)) = j$. Since each F^j is a sparse image, convolution can be applied efficiently in this case as well. Note that $\{F^j\}$ are mutually disjoint, thus the operations required for the k sparse convolutions sum to $O(mn)$ operations. As in the P2W case, the entire algorithm requires $O(mn + kn)$ operations, which is on the order of a single image convolution. Memory requirement is also economical and is on an order order of the image size.

Algorithm 2 MTM - Window-to-Pattern

```
{Input: pattern $\mathbf{p}$, image $F$.}
{Output: image $D$ of MTM distances.}
$p_1 := \mathbf{1} \cdot \mathbf{p}$   {pattern's sum}
$p_2 := \mathbf{1} \cdot (\mathbf{p} \odot \mathbf{p})$   {pattern's sum of squares}
$d_2 := p_2 - p_1^2/m$   {computation of $d_2$ (denominator)}
Generate $\{F^j\}$, for $j = 1..k$
$D_1 := 0$   {will accumulate the numerator}
for $j := 1$ to $k$ do
    $N := \mathbf{1} * F^j$   {calc $|\mathbf{w}_r^j| \in \mathbf{F}$}
    $T := flip(\mathbf{p}) * F^j$ {convolve image slice with $\mathbf{p}$}
    $T := (T \odot T) \oslash N$
    $D_1 := D_1 + T$
end for
$D := (p_2 - D_1)/d_2$
return D
```

3 Statistical Properties

In this section we give statistical justification for the proposed distance. We show that the MTM distance measures the relative reduction in the variance of the pattern given the window (or vice versa). With this rationalization, the MTM distance is similar in spirit to the MI measure where variance takes the role of entropy. We will show that the MTM is closely related to the Correlation Ratio measure [15] and to the Fisher Linear Discriminant [13]. Additionally, when restricting to linear tone mappings, the MTM minimization scheme results in the NCC solution. Thus, the MTM distance can be viewed as a generalization of the NCC distance when non-linear functional dependencies are considered.

Throughout this section we discuss the pattern-to-window case where $D(\mathbf{p}, \mathbf{w})$ distance is used. All observations and claims are obviously applicable in a symmetrical manner to the window-to-pattern case (using $D(\mathbf{w}, \mathbf{p})$).

Recall that $D(\mathbf{p},\mathbf{w})$ is composed of two terms:

$$D(\mathbf{p},\mathbf{w}) = \frac{d_1}{d_2}$$

where

$$d_1 = \|\mathbf{w}\|^2 - \sum_j \frac{1}{|\mathbf{p}^j|}(\mathbf{p}^j \cdot \mathbf{w})^2 \quad \text{and} \quad d_2 = m \cdot \text{var}(\mathbf{w})$$

Theorem 1 states that $D(\mathbf{p},\mathbf{w})$ measures the ratio between the conditional variance of $\mathbf{w}|\mathbf{p}$ and the variance of $\mathbf{w}$. As in Section 1, we consider the pattern $\mathbf{p}$ and the candidate window $\mathbf{w}$ as random variables with m samples p_i and w_i, $i = 1..m$, respectively.

Theorem 1.

$$D(\mathbf{p},\mathbf{w}) = \frac{E\left[\text{var}(\mathbf{w}|\mathbf{p})\right]}{\text{var}(\mathbf{w})}$$

where $E[\cdot]$ *is taken over all sample values* p_i *of* $\mathbf{p}$.

Proof 1. Proof is given in Appendix A.

Figure 4 shows a scatter diagram of a specific pair $\mathbf{p}$ and $\mathbf{w}$. The horizontal axis indicates pattern values and the vertical axis indicates corresponding window values. Each pair of values is represented as a point in the scatter diagram. The empirical mean $E[\mathbf{w}|\mathbf{p}^j]$ for the j^{th} bin is drawn as a full circle, and the conditional variance $\text{var}(\mathbf{w}|\mathbf{p}^j)$ is illustrated as a double headed arrow. Note, that in terms of MTM matching, $E[\mathbf{w}|\mathbf{p}^j]$ is the estimated tone map value for the tones in $\mathbf{p}$ associated with the j^{th} bin. The expectation value of $\text{var}(\mathbf{w}|\mathbf{p}^j)$ over all bins $\mathbf{p}^j$, $j = 1..k$ is $E[\text{var}(\mathbf{w} \mid \mathbf{p})]$. Intuitively, this evaluates the spread of the data around the estimated tone mapping.

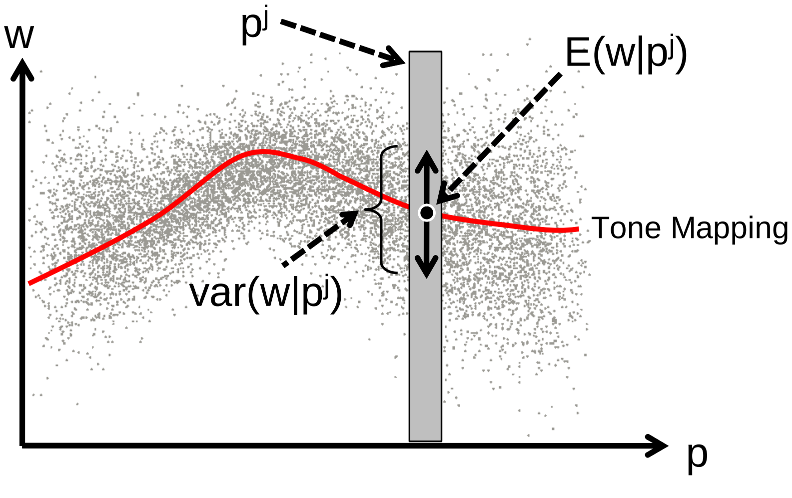

Figure 4: Conditional expectation $E[\text{var}(\mathbf{w}|\mathbf{p}^j)]$.

Thus, Theorem 1 implies that when seeking a good match for $\mathbf{p}$ over the entire image, a candidate window $\mathbf{w}$ is sought that minimizes the ratio $D(\mathbf{p},\mathbf{w}) = E[\mathrm{var}(\mathbf{w}|\mathbf{p})]/\mathrm{var}(\mathbf{w})$. Note, however, that rather than minimizing $D(\mathbf{p},\mathbf{w})$ one can equivalently maximize the term

$$\tilde{D}(\mathbf{p},\mathbf{w}) = 1 - D(\mathbf{p},\mathbf{w}) = \frac{\mathrm{var}(\mathbf{w}) - E[\mathrm{var}(\mathbf{w}|\mathbf{p})]}{\mathrm{var}(\mathbf{w})} \quad . \tag{13}$$

which is the normalized reduction in the variance of $\mathbf{w}$ when $\mathbf{p}$ is given. This relation bears a strong similarity to the mutual-information measure. In both cases the aim is to maximize the reduction in the *uncertainty* of $\mathbf{w}$ given $\mathbf{p}$. However, while in the MI scheme the entropy is used as the uncertainty measure, the MTM uses the variance as the uncertainty measure. Using variance rather than entropy enables the MTM scheme to be applied very fast on large images. Additionally, while the MI measure is very sensitive to the size of the bins (or the width of the kernel, if kernel estimation is used), the variance based MTM measure does not suffer from this parameter. Further discussion on MTM vs MI can be found in Section 5.

As stated above, the empirical mean $E[\mathbf{w}|\mathbf{p}^j]$ for the j^{th} bin (full circle in Figure 4) is the estimated tone-mapping for the values in $\mathbf{p}^j$. The collection $\{E[\mathbf{w}|\mathbf{p}^j]\}$ for all $j = 1..k$, forms the estimated optimal tone mapping that maps $\mathbf{p}$ to $\mathbf{w}$ (solid red curve in Figure 4). The variance of the collection $\{E[\mathbf{w}|\mathbf{p}^j]\}$ is closely related to the MTM distance $D(\mathbf{p},\mathbf{w})$. We state this relation in the following theorem:

Theorem 2.

$$\tilde{D}(\mathbf{p},\mathbf{w}) = 1 - D(\mathbf{p},\mathbf{w}) = \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{\mathrm{var}(\mathbf{w})}$$

Proof 2.

The theorem is derived directly from the *law of total variance* [30] which states:

$$\mathrm{var}(\mathbf{w}) = E[\mathrm{var}(\mathbf{w}|\mathbf{p})] + \mathrm{var}(E[\mathbf{w}|\mathbf{p}])$$

Therefore,

$$\tilde{D}(\mathbf{p},\mathbf{w}) = 1 - D(\mathbf{p},\mathbf{w}) = \frac{\mathrm{var}(\mathbf{w}) - E[\mathrm{var}(\mathbf{w}|\mathbf{p})]}{\mathrm{var}(\mathbf{w})} = \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{\mathrm{var}(\mathbf{w})} \qquad \square$$

Note that the term $\tilde{D}(\mathbf{p},\mathbf{w})$ is the Correlation Ratio statistical measure [15]. Roche et. al. suggested to use the Correlation Ratio measure for multi-modal image registration [29].

Theorem 2 implies that the mean of the conditional variance and the variance of the conditional mean are interchangeable, in the sense that, minimization over the first is the maximization over the latter:

$$\arg\min_{\mathbf{w}} \frac{E[\mathrm{var}(\mathbf{w}|\mathbf{p})]}{\mathrm{var}(\mathbf{w})} = \arg\max_{\mathbf{w}} \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{\mathrm{var}(\mathbf{w})}$$

Both measures are in the range $[0, 1]$. In the case where the optimal tone mapping is uniform, i.e. $\exists c,\ s.t.\ E(\mathbf{w}|\mathbf{p}^j) = c$ for all $j = 1..k$, then $\mathrm{var}(E[\mathbf{w}|\mathbf{p}]) = 0$ and $E[\mathrm{var}(\mathbf{w}|\mathbf{p})] = \mathrm{var}(\mathbf{w})$ and $\tilde{D}(\mathbf{p}, \mathbf{w}) = 0$ while $D(\mathbf{p}, \mathbf{w}) = 1$. Thus, although the $\mathbf{w}$ values are well predictable from $\mathbf{p}$, the MTM distance is still large since the predictability is only due to the low dispersion of $\mathbf{w}$. This property is imperative for pattern matching, since the $\mathbf{w}$ values located in smooth image regions (such as sky or non-textured surfaces) are predictable from $\mathbf{p}$, but this is not the desired matching solution. Note, that this is also the reason that the mutual information was preferred over the conditional entropy in [35].

Additionally, consider again the law of total variance [30]. It can be shown that

$$\arg\max_{\mathbf{w}} \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{\mathrm{var}(\mathbf{w})} = \arg\max_{\mathbf{w}} \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{E[\mathrm{var}(\mathbf{w}|\mathbf{p})]}$$

(See Appendix B) which yields that MTM is related to the Fisher Linear Discriminant [13], in which the goal is to maximize inter-class variance (numerator) while minimizing intra-class variance (denominator), where, in our case, each bin takes the role of a class.

In the final part of this section and for the sake of completeness we mention that the MTM scheme is a generalization of the NCC distance measure. In particular, when tone mappings are restricted to be linear functions, the NCC and MTM distances coincide [29]:

Theorem 3. *In the context of pattern matching using MTM, assume tone mappings are restricted to be linear, i.e.* $\mathcal{M}(\mathbf{p}) = a\mathbf{p} + b$, *where* a, b *are scalar parameters. Denoting by* $\rho(\mathbf{p}, \mathbf{w})$*, the normalized cross correlation distance as defined in Equation 1, we have:*

$$\tilde{D}(\mathbf{p}, \mathbf{w}) = \rho^2(\mathbf{p}, \mathbf{w})$$

Proof 3. Proof is given in Appendix C.

4 Piecewise Linear MTM

The benefits of using the piecewise constant (PWC) approximation for tone mappings as suggested in Section 2.1 are simplicity and computational efficiency. This approximation allows an unconstrained functional relationship between the pattern and the sought window. In some cases, however, this flexibility introduces a weakness as it generates an under-determined system of equations that may produce false matches. Such scenarios occur mainly when small sized patterns are sought and only sparse sampling values are available for many bins. Increasing the bin sizes of the SLT does not solve this problem as it also increases the mapping representation error. To allow larger bin sizes without degrading the modeling precision we extend the SLT transform to implement a higher order model, namely, a *Piecewise Linear* (PWL) approximation. This representation approximates the tone mapping as a piecewise linear function and enables aggregating more samples for each bin without compromising representation accuracy. Similar to the Piecewise Constant SLT, the PWL-SLT slices a given image into k slices, but rather than being binary slices, the slices now contain real values.

Recall the SLT definition described in Section 2.1, we denote bin boundaries as a sequence $q_1, \cdots, q_{k+1}$ where $q_1 < q_2 < \cdots < q_{k+1}$. A value x in the half open interval $[q_1, q_{k+1})$ is associated with a bin $\pi(x)$

$$\pi(x) = j \quad \text{if} \;\; x \in [q_j, q_{j+1})$$

We define $r(x)$ to be the relative position of x in its bin:

$$r(x) = \frac{x - q_{\pi(x)}}{q_{\pi(x)+1} - q_{\pi(x)}}$$

Note, that $r(x) \in [0, 1]$, where $r(x) = 0$ if $x = q_{\pi(x)}$, and $r(x) \to 1$ when $x \to q_{\pi(x)+1}$. For every $x \in [q_1, q_{k+1})$ the following relation holds:

$$x = (1 - r(x)) \cdot q_{\pi(x)} + r(x) \cdot q_{\pi(x)+1} \tag{14}$$

Defining a $k + 1$ dimensional vector α as a vector composed of the bin boundaries:

$$\alpha = [q_1, q_2, \cdots, q_{k+1}]$$

Equation 14 can be rewritten in vectorial form:

$$x = Q(x)\alpha \tag{15}$$

where $Q(x)$ is a row vector:

$$Q(x) = [0, \cdots, 0, 1 - r(x), r(x), 0, \cdots, 0]$$

s.t. the values $1-r(x)$ and $r(x)$ are located in the $\pi(x)$ and $\pi(x)+1$ entries, respectively. We now define a matrix extension of Equation 15. Let $\mathbf{x} \in \mathcal{R}^m$ be a real valued vector whose elements satisfy $x_i \in [q_1, q_{k+1})$. The *piecewise linear slice transform* (PWL-SLT) of $\mathbf{x}$ is defined as:

$$\mathbf{x} = Q(\mathbf{x})\alpha \tag{16}$$

where $Q(\mathbf{x})$ is an $m \times (k + 1)$ SLT matrix:

$$[Q(\mathbf{x})](i, j) = \begin{cases} 1 - r(x_i) & \text{if} \;\; \pi(x_i) = j \\ r(x_i) & \text{if} \;\; \pi(x_i) = j - 1 \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

Note that, in contrast with the PWC case, multiplying $Q(\mathbf{x})$ with the vector α does not quantize $\mathbf{x}$ but reproduces $\mathbf{x}$ exactly (Equation 16), regardless of the number of bins. Substituting the boundary vector in the expression $Q(\mathbf{x})\alpha$ with a different vector β we obtain a piecewise linear tone mapping of $\mathbf{x}$:

$$\mathbf{y} = Q(\mathbf{x})\beta \tag{18}$$

This mapping implies that values in the interval $[\alpha_i, \alpha_{i+1})$ are linearly mapped to the interval $[\beta_i, \beta_{i+1})$. Note, that in contrast with the PWC-SLT matrix $S(\mathbf{x})$, the columns of matrix $Q(\mathbf{x})$ are not orthogonal. Thus we define a variant of the original image slice defined in Section 2.1:

We define $\tilde{\mathbf{x}}^j = [\tilde{x}^j_1, \cdots, \tilde{x}^j_m]$ as a real valued vector associated with the j^{th} bin:

$$\tilde{x}^j_i = \begin{cases} r(x_i) & \text{if } \pi(x_i) = j \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

The matrix $Q(\mathbf{x})$ can then be represented as a collection of column vectors (slices):

$$Q(\mathbf{x}) = [\bar{\mathbf{x}}^1, \bar{\mathbf{x}}^2, \cdots, \bar{\mathbf{x}}^{k+1}]$$

where we define

$$\bar{\mathbf{x}}^j = \mathbf{x}^j - \tilde{\mathbf{x}}^j + \tilde{\mathbf{x}}^{j-1} \tag{20}$$

where $\tilde{\mathbf{x}}^j$ is defined above (Equation 19), and $\mathbf{x}^j$ is the originally defined slice vector (Equation 3). The end cases are set to be: $\mathbf{x}^{k+1} = \tilde{\mathbf{x}}^{k+1} = \tilde{\mathbf{x}}^0 = \mathbf{0}$.

In the following sections we describe how MTM-PWL matching can be applied efficiently on the entire image. In this case, the minimum normed distance between a pattern and candidate window (Equations 1 and 2) is evaluated under all possible *piecewise linear* tone mappings. Again, since tone mapping is not necessarily a bijective mapping, two alternatives must be considered: Pattern-to-Window (P2W) and Window-to-Pattern (W2P).

4.1 P2W by Piecewise Linear Mapping

The MTM distance is given by the minimization of Equation 1 where $\mathcal{M}(\mathbf{p}) = Q(\mathbf{p})\beta$. The optimal mapping is then given by:

$$\hat{\beta} = \arg\min_{\beta} \|Q\beta - \mathbf{w}\|^2 = (Q^TQ)^{-1}Q^T\mathbf{w}$$

For simplicity we denote $Q(\mathbf{p})$ by Q. Substituting back into Equation 1, the MTM distance reads:

$$D(\mathbf{p}, \mathbf{w}) = \frac{\| Q\hat{\beta} - \mathbf{w}\|^2}{m \cdot \mathrm{var}(\mathbf{w})} = \frac{\|Q\left(Q^TQ\right)^{-1}Q^T\mathbf{w} - \mathbf{w}\|^2}{m \cdot \mathrm{var}(\mathbf{w})} \tag{21}$$

Expanding the numerator of the above term we obtain:

$$d_1 = \|Q\left(Q^TQ\right)^{-1}Q^T\mathbf{w} - \mathbf{w}\|^2 = \|\mathbf{w}\|^2 - (\mathbf{w}^TQG^{-1}Q^T\mathbf{w}) \tag{22}$$

where $G = Q^TQ$. Unlike the PWC case, the matrix G in this case is not diagonal, thus calculating the right hand term in Equation 22 is more challenging than the PWC case. Fortunately, G is a tridiagonal matrix (see Appendix D), a property we exploit to expedite calculations. Note, that in the P2W case, G is a function of $\mathbf{p}$ and may be calculated only once for all candidate windows.

Consider the right most term in Equation 22: $\mathbf{w}^TQG^{-1}Q^T\mathbf{w}$, and recall that the columns $\bar{\mathbf{p}}^j$ of Q are given by Equations 19 and 20. Thus calculating $Q^T\mathbf{w}$ requires $2k$ dot products $\{\mathbf{w}\cdot\mathbf{p}^j\}_{j=1}^k$ and $\{\mathbf{w}\cdot\tilde{\mathbf{p}}^j\}_{j=1}^k$:

$$\rho_j = [Q^T\mathbf{w}]_j = \mathbf{w}\cdot\bar{\mathbf{p}}^j = \mathbf{w}\cdot(\mathbf{p}^j - \tilde{\mathbf{p}}^j + \tilde{\mathbf{p}}^{j-1})$$

However, since the pattern slices are mutually exclusive, the k dot-products with $\{\mathbf{p}^j\}$ as well as with $\{\tilde{\mathbf{p}}^j\}$ require only $O(m+k)$ operations, for each.

Calculating the entire term $\mathbf{w}^TQG^{-1}Q^T\mathbf{w}$ requires additional multiplication of $Q^T\mathbf{w}$ with G^{-1}. Since G^{-1} is a $k\times k$ matrix, this would require an additional k^2 operations. However, since G is tridiagonal we use the *Tridiagonal Matrix Algorithm* (TDMA) [10] in the following manner. Denote $\hat{\beta} = G^{-1}Q^T\mathbf{w}$, thus $G\hat{\beta} = Q^T\mathbf{w}$. Using TDMA, solving for $\hat{\beta}$ can be implemented in $O(m+k)$ operations using Gaussian elimination and backward substitution (Appendix E). Therefore, calculating the entire term $\mathbf{w}^TQG^{-1}Q^T\mathbf{w}$ requires $O(2(m+k))$ operations. Algorithm 3 gives a pseudo-code for applying P2W pattern matching over an entire image F using PWL approximation. In the pseudo-code capital letters and bracketed variables (such as $[\rho_j]$) represent images of size equal to F. Assuming the image F is of n pixels, the entire search requires $O(2(nm+nk))$ operations, which is equivalent to two image convolutions!

4.2 W2P by Piecewise Linear Mapping

Due to the symmetry between $\mathbf{p}$ and $\mathbf{w}$ we can exchange $\mathbf{p}$ and $\mathbf{w}$ in Equations 21 and 22 above and obtain:

$$D(\mathbf{w},\mathbf{p}) = \frac{\|\,Q\hat{\beta} - \mathbf{p}\|^2}{m\cdot\mathrm{var}(\mathbf{p})} = \frac{\|\mathbf{p}\|^2 - \mathbf{p}^TQG^{-1}Q^T\mathbf{p}}{m\cdot\mathrm{var}(\mathbf{p})}$$

where $Q = Q(\mathbf{w})$ and $G = Q^TQ$. The scalars $\|\mathbf{p}\|^2$ and $\mathrm{var}(\mathbf{p})$ are calculated once for all windows, however, the term $\mathbf{p}^TQG^{-1}Q^T\mathbf{p}$ must now be calculated explicitly for each window in F. In this case, we denote

$$\rho_j = [Q^T\mathbf{p}]_j = \mathbf{w}^j\cdot\bar{\mathbf{p}}$$

Algorithm 3 MTM-PWL: Pattern-to-Window

```
{Input: pattern $\mathbf{p}$, image $F$. Output: image $D$ of distances}
{Calculate PWL Pattern Slices}
Generate $\{\mathbf{p}^j\}_{j=1}^k$
Generate $\{\tilde{\mathbf{p}}^j\}_{j=1}^k$
$\mathbf{p}^0 = \mathbf{p}^{k+1} = \tilde{\mathbf{p}}^0 = \tilde{\mathbf{p}}^{k+1} = \mathbf{0}$
$\bar{\mathbf{p}}^j = \mathbf{p}^j - \tilde{\mathbf{p}}^j + \tilde{\mathbf{p}}^{j-1}$, for $j = 1..k+1$

{Calculate matrix $G$ according to Eq. 35 Appendix D}
$\varphi_p^j = \bar{\mathbf{p}}^j \cdot \bar{\mathbf{p}}^j$, for $j = 1..k+1$
$\psi_p^j = \bar{\mathbf{p}}^j \cdot \bar{\mathbf{p}}^{j+1}$, for $j = 1..k$
Calculate $\omega_p^j$, for $j = 1..k$   {Eq. 38 Appendix E}

{Calculate all window projections}
$\tilde{T}^0 = \mathbf{0}$;
$T^j = flip(\mathbf{p}^j) * F$, for $j = 1..k+1$   {calculate $\mathbf{p}^j \cdot \mathbf{w}, \ \forall \mathbf{w} \in F$}
$\tilde{T}^j = flip(\tilde{\mathbf{p}}^j) * F$, for $j = 1..k+1$   {calculate $\tilde{\mathbf{p}}^j \cdot \mathbf{w}, \ \forall \mathbf{w} \in F$}

{TDMA - Forward pass}
for $j := 1$ to $k+1$ do
    $[\rho_j] = T^j - \tilde{T}^j + \tilde{T}^{j-1}$   {Compute $[Q^T\mathbf{w}]_j$}
    Calculate $[\sigma_j]$   {Eq. 39 Appendix E}
end for

{TDMA - Backward pass}
for $j := k+1$ to $1$ do
    Calculate $[\hat{\beta}_j]$   {Eq. 40 Appendix E}
end for

{Calculate distances for all windows}
$D_1 = \sum_j [\hat{\beta}_j] \odot [\rho_j]$   {calculation of $d_1$ (numerator)}
$W_1 := \mathbf{1} * F$   {window sum}
$W_2 := \mathbf{1} * (F \odot F)$   {window sum of squares}
$D_2 := W_2 - (W_1 \odot W_1)/m$   {calculation of $d_2$ (the denominator)}
$D := (W_2 - D_1) \oslash D_2$
return D
```

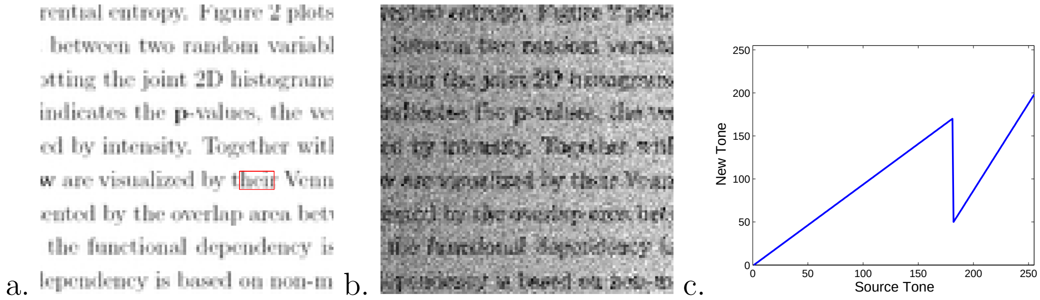

Figure 5: (a) Original image with a sub-window used as pattern (marked square). (b) Tone mapped image with added noise. (c) Non-Monotonic mapping used to create (b).

where

$$\bar{\mathbf{w}}^j = \mathbf{w}^j - \tilde{\mathbf{w}}^j + \tilde{\mathbf{w}}^{j+1}$$

We use again the tridiagonal property of G and the TDMA algorithm to produce the MTM distance for each window in image F with $O(2(nk + nm))$ operations. Algorithm 4 describes the process of calculating $D(\mathbf{w}, \mathbf{p}), \forall \mathbf{w} \in F$. In this algorithm we define F^j to be the j^{th} slice of image F, i.e. $F^j(x, y) = 1 \Leftrightarrow \pi(F(x, y)) = j$, and $\tilde{F}^j$ to be the j^{th} PWL image slice, i.e. $\tilde{F}^j(x, y) = r(F(x, y)) \Leftrightarrow \pi(F(x, y)) = j$.

5 Results

The suggested method was tested under various scenarios. Performance was compared with four distance measures discussed in Section 1, namely, the Euclidean distance, Local Binary Pattern (LBP), Normalized Cross Correlation (NCC), and Mutual Information (MI). In this section we show that even under extreme tone mappings and under heavy noise conditions, the MTM approach successfully and efficiently detects the sought patterns, performing significantly better than the compared methods.

To illustrate the performance of the proposed approach, consider the image and the selected pattern presented in Figure 5a. Figure 5b shows the original image after applying the tone mapping plotted in Figure 5c, adding a tone gradient and adding white Gaussian noise. The selected pattern was sought in the tone mapped image by evaluating the distance measures between the pattern and every window in the image using the five distances mentioned above. For MI and MTM measures, bin size was chosen to provide best results (MTM with bin size 20, MI with bin size 40).

Figure 6 presents the distance values for all image windows for each of the measures, shown as a color map. The maps in Figures 6d-e have been morphologically eroded to assist in visualizing the sharp detection peaks (these peaks are in fact much sharper). From the maps it can be seen that the MTM distance (Figure 6e) clearly shows a sharp

Algorithm 4 MTM-PWL: Window-to-Pattern

```
{Input: pattern $\mathbf{p}$, image $F$. Output: image $D$ of distances}
{Calculate PWL Window Slices}
Generate $\{\mathbf{F}^j\}_{j=1}^k$
Generate $\{\tilde{\mathbf{F}}^j\}_{j=1}^k$
$\mathbf{F}^0 = \mathbf{F}^{k+1} = \tilde{\mathbf{F}}^0 = \tilde{\mathbf{F}}^{k+1} = \mathbf{0}$
$\bar{F}^j = F^j - \tilde{F}^j + \tilde{F}^{j-1}$, for $j = 1..k+1$

{Calculate matrix $G$ for each window (Eq. 35)}
$[\varphi_w^j] = \mathbf{1} * (\bar{F}^j \odot \bar{F}^j)$, for $j = 1..k+1$   {calculates $\bar{\mathbf{w}}^j \cdot \bar{\mathbf{w}}^j$, $\forall \mathbf{w} \in F$}
$[\psi_w^j] = \mathbf{1} * (\bar{F}^j \odot \bar{F}^{j+1})$, for $j = 1..k$   {calculates $\bar{\mathbf{w}}^j \cdot \bar{\mathbf{w}}^{j+1}$, $\forall \mathbf{w} \in F$}
$[\omega_w^j]$ for $j = 1..k$  {according to Eq. 38 Appendix E}

{Calculate all pattern projections}
$\tilde{T}^0 = \mathbf{0}$;
$T^j = \mathbf{p} * F^j$, for $j = 1..k+1$   {calculate $\mathbf{p} \cdot \mathbf{w}^j$, $\forall \mathbf{w} \in F$}
$\tilde{T}^j = \mathbf{p} * \tilde{F}^j$, for $j = 1..k+1$   {calculate $\mathbf{p} \cdot \tilde{\mathbf{w}}^j$, $\forall \mathbf{w} \in F$}

{TDMA - Forward pass}
for $j := 1$ to $k+1$ do
  $[\rho_j] = T^j - \tilde{T}^j + \tilde{T}^{j-1}$   { Compute $[Q^T\mathbf{p}]_j$}
  Calculate $[\sigma_j]$   {Eq. 39 Appendix E }
end for

{TDMA - Backward pass}
for $j := k+1$ to 1 do
  Calculate $[\hat{\beta}_j]$   {Eq. 40 Appendix E}
end for

{Calculate distances for all windows}
$D_1 = \sum_j [\hat{\beta}_j] \odot [\rho_j]$   {calculation of $d_1$ (numerator)}
$p_1 := \mathbf{1} \cdot \mathbf{p}$   {pattern sum}
$p_2 := \mathbf{1} \cdot (\mathbf{p} \odot \mathbf{p})/m$   {pattern sum of squares}
$d_2 := p_2 - p_1^2$   {calculation of $d_2$ (the denominator)}
$D := (p_2 - D_1)/d_2$
return D
```

peak at the correct location overcoming both non-monotonic mapping and noise. The Euclidean and the LBP measures both strongly fail due to the non linearity of the mapping and due to the noise. The NGC, fails due to the non-linearity of the tone mapping. The MI shows confusion in the detection locations, this is mainly due to the relatively small pattern size which implies very sparse data in the MI bins (even when bin size increases to 40 gray values).

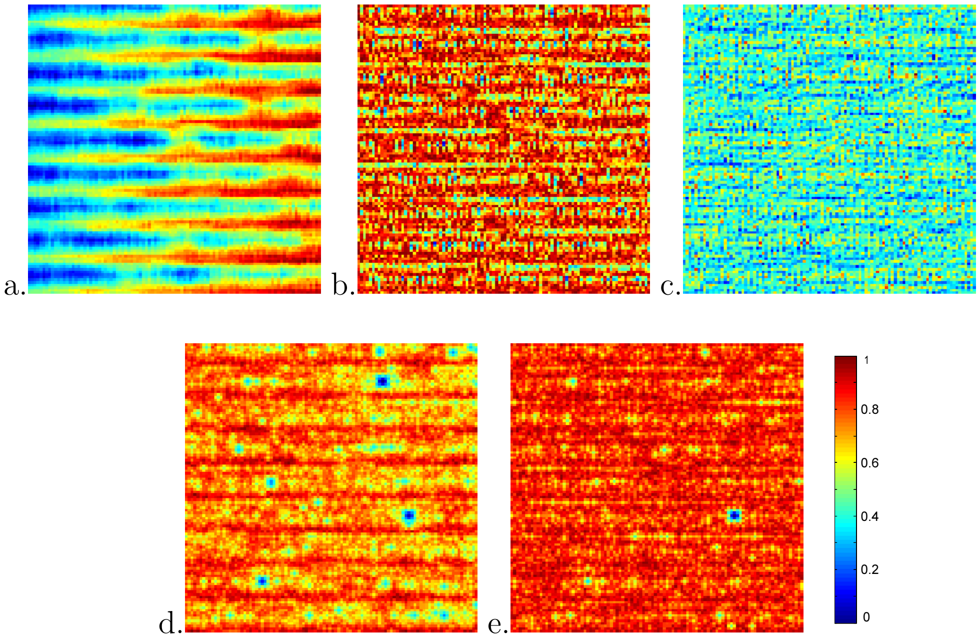

Figure 6: The pattern shown in Figure 5a was sought in the tone mapped image of Figure 5b. Distance maps display distance values between pattern and image windows for different distance measures. (a) Euclidean (b) NGC (c) LBP (d) MI (e) MTM.

5.1 Sensitivity to Mapping Extremity

In order to quantitatively evaluate the performance of the suggested approach and compare it with other methods we conducted a series of experiments on gray-level images. Pattern matching was applied on a large set of randomly selected image-pattern pairs under various conditions. For each input image, a pattern of a given size was selected at a random location. To prevent smooth or "non-interesting" patterns, these were selected from among patterns with a degree of "structure" above a specific threshold. "Structure" was measured at each pixel as the direct sum of the structure tensor's eigen-values (a 2x2 matrix of partial derivatives [5]). Given an image and a selected pattern, a random tone mapping was applied to the image (with additive noise) and the original selected pattern was then sought in the mapped image. Distances were

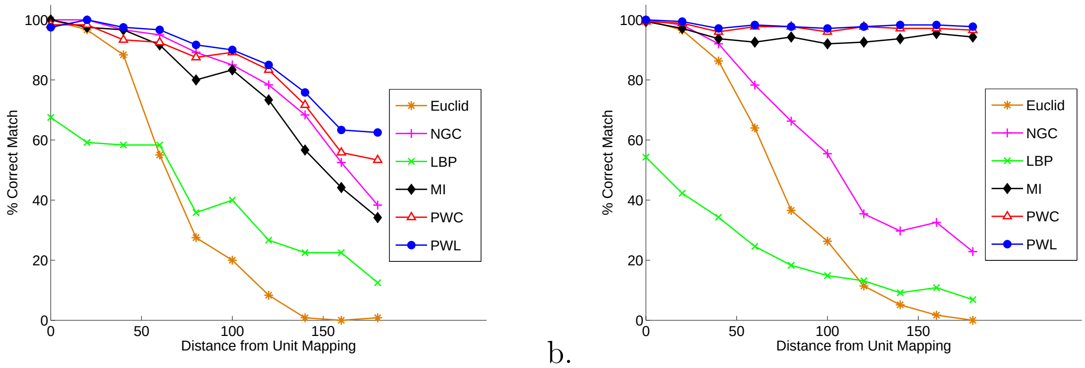
a. b.

Figure 7: Pattern detection performance vs. extremity of the tone mapping for (a) monotonic mapping with noise (b) non-monotonic mapping with noise.

calculated for all possible locations in the mapped image, and the window associated with the minimal distance was considered the *matched window*. If the matched window was detected at the correct position the match is considered a *correct detection*[3].

Figure 7 displays the detection rate as a function of the extremity of the tone mapping applied to the image. Extremity of tone mapping was measured as the mean squared distances between the original range of values ([0..255]) and the mapped tone values. Results are shown separately for monotonic mappings (Figure 7a) and for non-monotonic mappings (Figure 7b). Each data point represents the detection rate (in percentages) over 2000 randomly selected image-pattern pairs. Images were of size 200×200 and patterns of size 20×20. Tone mappings were generated by randomly selecting six new tone values serving as the mapping values for six equally spaced source tone values (in the range [0..255]). The tone mapping was defined as a piecewise linear function passing through the selected values. For monotonic mappings the randomly selected tone values were sorted in increasing order prior to the construction of the tone mapping. Additive Gaussian noise with s.t.d. of 15 gray-values was added to each mapped image before pattern matching was performed.

We note an important caveat with respect to the mappings and their extremity measure: in the case of monotonic mappings, the monotonicity constrains the extreme mappings and typically produces deeply convex or concave mappings which imply loss of spatial details in certain image regions. This in turn increases the difficulty of correctly detecting a given pattern. Non-monotonic mappings on the other hand, produce false contours but typically maintain the image structure (preserve original

---

[3] We note that using the top 5 or 10 minimal distance windows, did not significantly change the results.

edges though possibly increasing or decreasing their contrast). Thus, as will be seen, performance under monotonic mappings is often degraded compared to non-monotonic mappings.

Figure 7 shows that the Euclidean distance and the LBP degrade very fast with the mapping extremity. Indeed, this is expected for the Euclidean case, however, the LBP shows poor performance also in monotonic mappings where it should be insensitive to such mappings. This can be explained by noise that was added to the image and, as will be shown below, the LBP is very sensitive to additive noise. The NGC is expected to fail in both monotonic and non-monotonic mappings, however in the monotonic case, mapping is smooth and can be approximated locally as linear. Thus, NGC performs relatively well under monotonic mappings compared to the non-monotonic mappings.

It can be seen that the MTM approach in both PWC (red lines-unfilled markers) and PWL (blue lines - solid markers) schemes, performs very well and on par with the MI approach. Both, MTM and MI perform better under non-monotonic mappings compared with monotonic mappings due to the caveat described above. The MTM and MI methods were optimized for bin size (in this case MTM used bin size 40 and MI bin size 20).

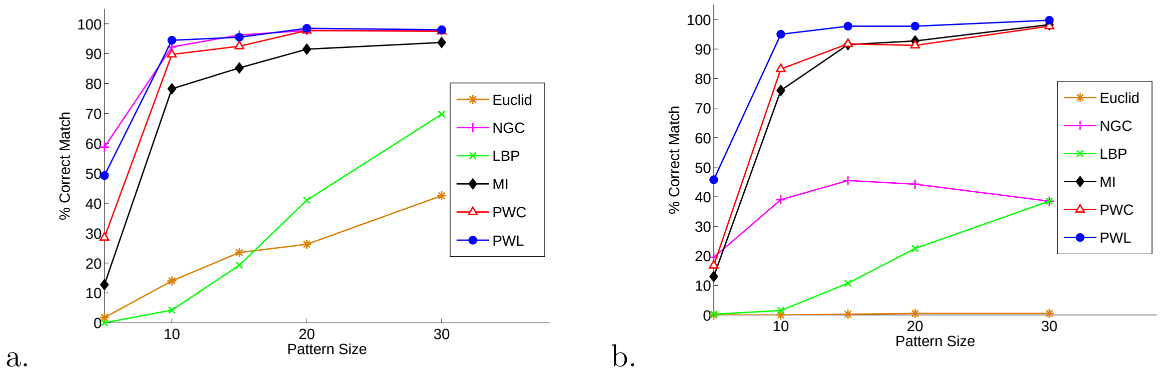

Figure 8: Performance comparison as a function of pattern size. (a) For specific monotonic mapping. (b) For specific non-monotonic mapping. Gaussian noise of std=15 was added in both cases.

5.2 Sensitivity to Noise and Pattern Size

We examined the sustainability of the mentioned distances to additive noise and its performance under various pattern sizes. Figure 8 shows the detection rate for various pattern sizes under a specific monotonic mapping (Figure 8a) and under a specific non-monotonic mapping (Figure 8b). All images were contaminated with Gaussian noise

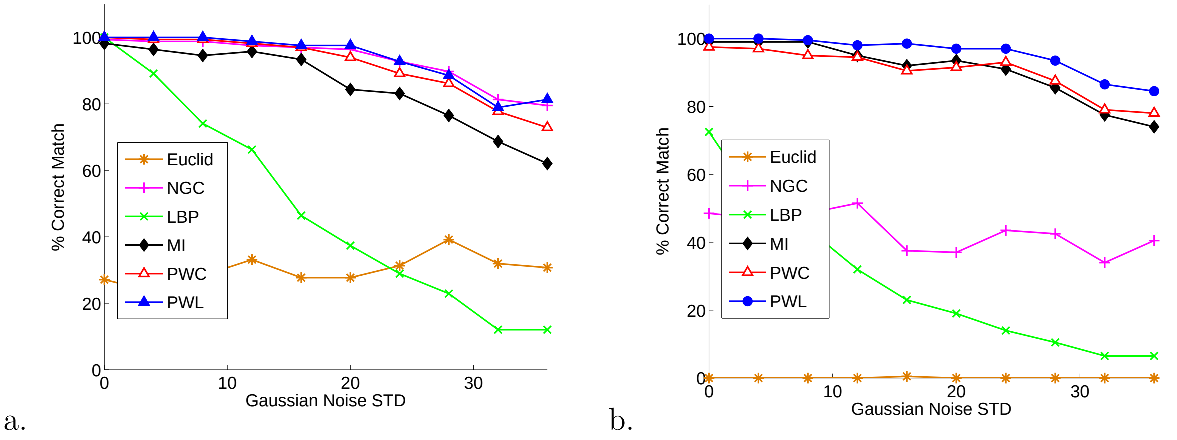

Figure 9: Performance comparison as a function of added noise. (a) For specific monotonic mapping. (b) For specific non-monotonic mapping.

with s.t.d. $= 15$. It can be seen that for small patterns (under 10×10 pixels) detection rates are very low in all methods. This behavior stems from the fact that histogram bins of small sized patterns are populated with sparse samples or even not populated at all. This may produce an under-determined system or an over-fitting solution. For this reason techniques using a low number of free parameters are preferable and outperform other methods in small pattern scenarios (NGC for monotonic mappings and MTM-PWL). On the other hand, techniques with a small number of free parameters do not model well all possible tone mappings (NGC for non-monotonic mappings). Figure 8 shows that MTM-PWL outperforms all other methods. This stems from the fact that the PWL model allows increasing the bin size without sacrificing the mapping accuracy. This implies lowering of the number of free parameters, and aggregating more pixel values in each bin. In turn, this increases stability of the solution and reduces over-fitting phenomena. This phenomena is also shown below in Figure 10.

Figure 9 evaluates the sensitivity of the above methods to additive noise. Pattern matching was performed under a specific mapping with Gaussian noise of varying variance added to each image. As above, data points represented average results over 2000 randomly selected image-pattern pairs. Figures 9a and 9b plot the results for monotonic and non-monotonic mappings respectively. Overall, the results resemble the behavior shown above in Figure 8. Methods with a small number of free parameters perform better, as long as they model well the possible tone-mappings. It can be seen that in both cases MTM-PWL is advantageous over MTM-PWC especially under severe noise conditions.

Finally, we test for the sensitivity of the MTM to bin size and compare with that of

MI. Figure 10 shows detection rates for MTM-PWC, MTM-PWL and MI over different bin sizes. Results are shown for three different pattern sizes (8×8, 16×16 and 32×32). Each data point is a result of 200 randomly selected image-pattern pairs. For every image-pattern pair, a random monotonic mapping was generated, within the extremity range of 40-60, and Gaussian noise (s.t.d = 20) was added. These plots show the difference in sensitivity to bin-size between the approaches. As expected, MTM-PWL outperforms MTM-PWC accross pattern sizes as well as MI, and is especially advantageous when using large bin size on smaller patterns. MI shows larger sensitivity to bin size with decrease in performance for smaller bin sizes.

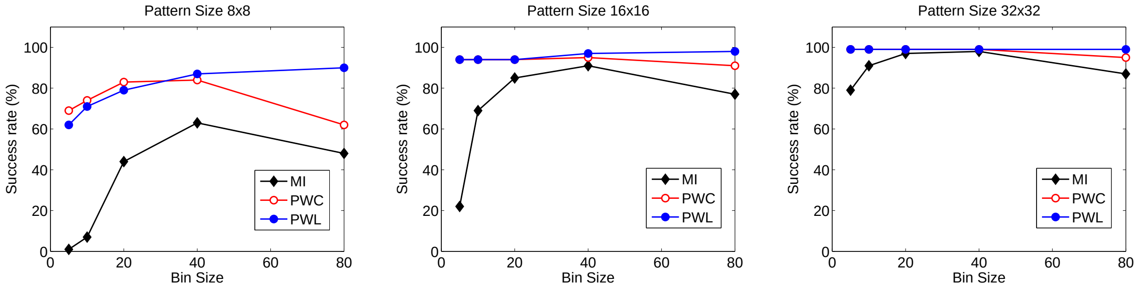

Figure 10: Performance comparison of MI and MTM as a function of bin size. For patterns of size 8×8, 16×16 and 32×32.

5.3 Run Time

A significant advantage of MTM over MI is computational efficiency. Figure 11 displays run times of pattern matching using different schemes under varying pattern sizes. For MI and MTM, run times are shown for different bin sizes as well. Run times shown are the average over 10 runs. Run time was measured on an Intel 1.70 GHz Pentium M. Since MI requires the computation of the joint histogram for every image window pairs, it is more computationally demanding than MTM and other approaches. Furthermore, run time for MI increases with the number of bins. On the other hand, run time of the MTM-PWC scheme is of the order equivalent to a single image convolution (Section 4.1) and thus on par with the NCC and EUC approaches. Run time of the MTM-PWL scheme is slightly higher than the MTM-PWC (two image convolutions). The size of bins in both, MTM-PWC and MTM-PWL, has very little effect on the run time.

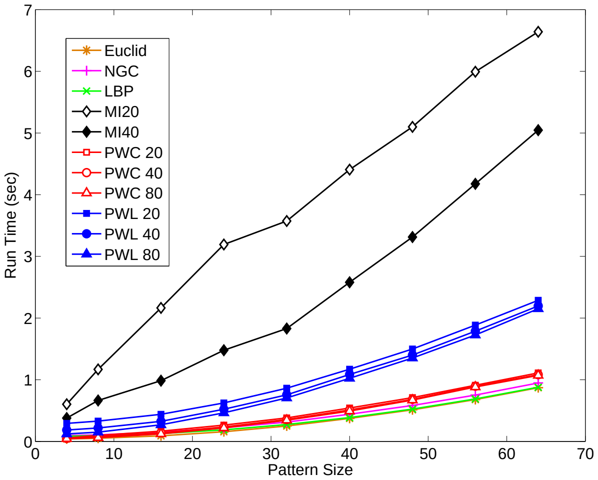

Figure 11: Algorithm run time for pattern matching under various schemes (and different bin sizes) for different pattern sizes.

6 Additional Applications

6.1 MTM for Multi-Modal Image Registration

To compare the capabilities of the MTM and MI in multi modality scenarios we evaluated the performance of pattern matching and image alignment between image pairs originating from different modalities, including: visual - SAR, visual - InfraRed, CT - MRI and CT - PET. These images display tone changes between their counterparts which are one-to-many, thus, cannot be regarded as a mapping function.

Figure 12 displays the detection rates of pattern matching between multi-modal images for various pattern sizes. The recorded performance is an average over five multi-modal image pairs where 100 (randomly selected) pattern-window cases were

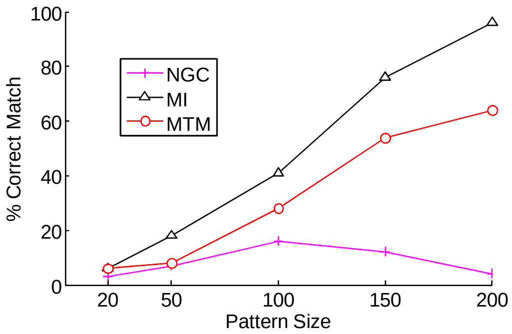

Figure 12: Detection rates of pattern matching between multi-modal images for various pattern sizes.

tested for each pair. It is demonstrated that MI outperforms MTM and NGC for almost all pattern sizes. While NGC severely fails in almost all pattern sizes, MTM presents reasonable performance although inferior to MI. MTM, however is significantly faster than MI as discussed above.

With respect to multi-modal image registration, both MI and MTM present comparable accuracies since the matched areas are extremely large. To demonstrate this, we evaluated the distance between an image and its modality counterpart under different translation parameters. Figure 13 displays distance maps between pairs of images of different modalities including visual-IR, visual-SAR and CT-MRI. Distance maps for NGC, MTM and MI are shown (left to right). In each map, the center of the map corresponds to the correct translation ($\Delta x = 0, \Delta y = 0$) and distance values for other translation parameters (sampled in pixel's width steps) are represented at the corresponding locations. Map values are represented in pseudo-color corresponding to the color-bar shown on the right. As can be seen, the minimum distance in the MI and MTM maps correspond to the correct translation with a deep and global minima. The results show that MTM and MI are comparable in their accuracy of alignment whereas NGC largely fails. Similar performance is observed for other image modalities as well.

Similar results are obtained for distance maps between these pairs of images for rotation and scale transformation as shown in Figure 14. Rotation parameters (x-axis of map) range from $-90°$ to $+90°$. Scale parameters (y-axis of map) range from 0.8 to 1.2. As above, the center of the map corresponds to the correct transformation parameters ($\Delta\theta = 0, s = 1$).

Note that, searching for the correct transformation parameters in a multi-modal alignment using MTM, can be implemented very efficiently: image slices need to be computed only once for the reference image of the pair, while transformed image requires only resampling and pointwise multiplication with the image slices. In contrast, searching for the transformation parameters using MI requires computation of the joint histogram of the image pairs for each candidate parameter - a time consuming process.

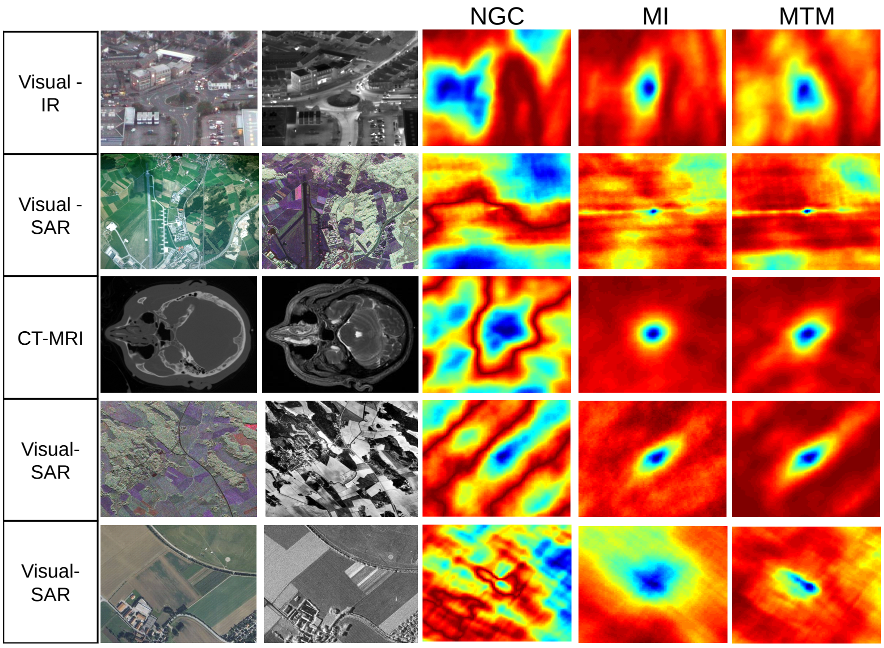

Figure 13: Distance maps between multi-modal image pairs for horizontal and vertical translations. Columns left to right: image pairs, distance maps for NGC, MI and MTM.

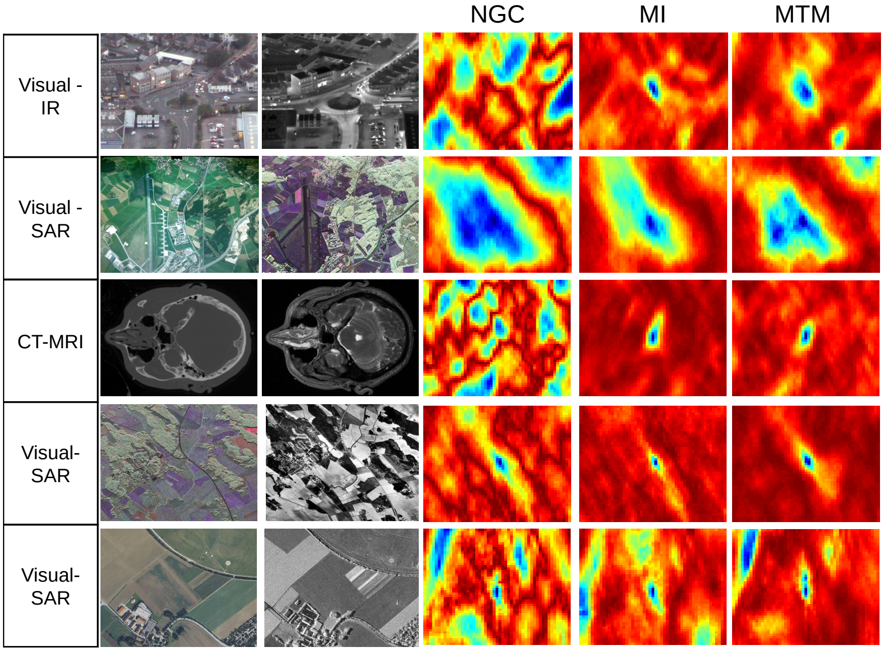

Figure 14: Distance maps between multi-modal image pairs for rotation and scale. Columns left to right: image pairs, distance maps for NGC, MI and MTM. Rotation parameters (x-axis of map) range from $-90°$ to $+90°$. Scale parameters (y-axis of map) range from 0.8 to 1.2.

6.2 Foreground Detection in Video

As an example of implementation of the MTM in a real time scenario, moving shadows were detected in video clips acquired by outdoor surveillance cameras [8]. These outdoor sequences show pedestrians walk across a static background with shadows varying in size and length dependent on the time of day (see Figure 15). With the goal of detecting pedestrians and not their shadows, recent approaches to shadow removal have difficulty in such cases since shadows are non uniform, noisy and tend to have wide penumbras [1]. Following classic techniques, a background image is computed and continuously updated by the system using a Mixture of Gaussians approach [26]. For every frame of the sequence, the foreground regions are determined using background subtraction. These regions include the pedestrians as well as their shadows (see Figure 15).

To distinguish between pedestrians and shadows, we exploit the assumption that the shadow regions are a (not necessarily linear) tone mapping of the background. Thus, the MTM distance is computed between the spatial neighborhood of every pixel within the detected foreground regions and the corresponding region in the background

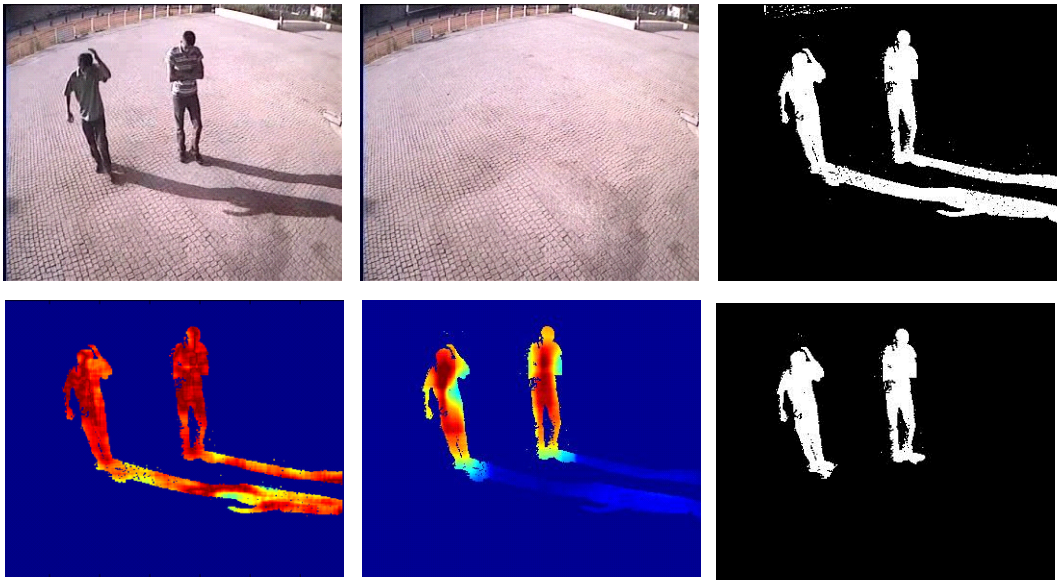

Figure 15: Top: A video frame acquired from a low-cost surveillance camera (left) and the current background image (middle). Foreground detection using background subtraction includes both actual foreground objects and shadows (right). Bottom: Distance maps obtained for the foreground pixels using NCC (left) and MTM (middle). Bright colors represent high values while dark colors represent low values. Right: Thresholding of the MTM distances results in foreground image without shadows.

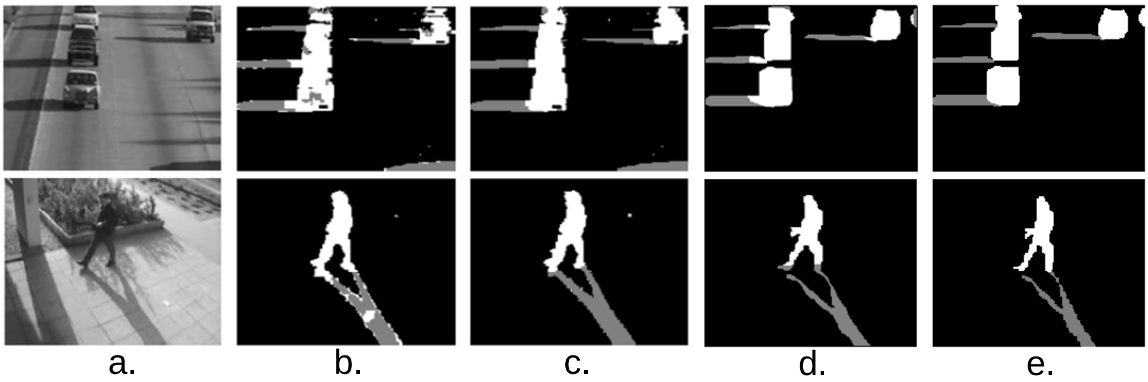

Figure 16: Shadow detection results. (a) Frames from the Highway (top) and Seam (bottom) sequences, their shadow detection results using (b) the Constant Ratio (CR) model, (c) the Statistical Shadow (SS) model, and (d) the proposed MTM method. (e) Displays the ground-truth. Detected shadow pixels are shown in gray and foreground pixels in white. Images are form [8].

image. Figure 15 (bottom) depicts the distance map calculated for the video frame shown in Figure 15 - top) using NCC (left) and MTM (middle). The MTM results in substantially lower values for shadowed pixels compared to foreground pixels. This result is compared with the Normalized Cross-Correlation (NCC) metric that fails to do so due to its false assumption of linear mapping between shadowed pixels and their corresponding background pixels. Shadows are detected by thresholding the MTM distance map using a two-class thresholding method [25]. Resulting foreground image without shadows is shown in Figure 15 (bottom-right).

To evaluate performance, the proposed approach was compared in [8] with two shadow detection methods - Constant Ratio (CR) [33] and Statistical Shadow (SS) [4]. The SZTAKI benchmark set [4, 28] was used as input. This dataset contains video sequences with moving shadows in different scenarios, all of which have ground truth, and results of the CR and SS methods. Figure 16 shows two video frames from the dataset and their corresponding foreground and shadow detection results according to the three shadow detection methods. It can be seen that the MTM's results are smoother and more accurate with respect to the given ground-truth. Figure 17 compares quantitative shadow detection results of these methods using the F-measure (the harmonic mean of precision and recall) over all frames of the sequence. The results show that the proposed technique substantially outperforms the CR and SS methods, yielding a significant advantage in terms of the F-measure. The proposed technique results in about 21% increase in the F-measure compared with the CR method and about 7% increase compared with the SS method on average.

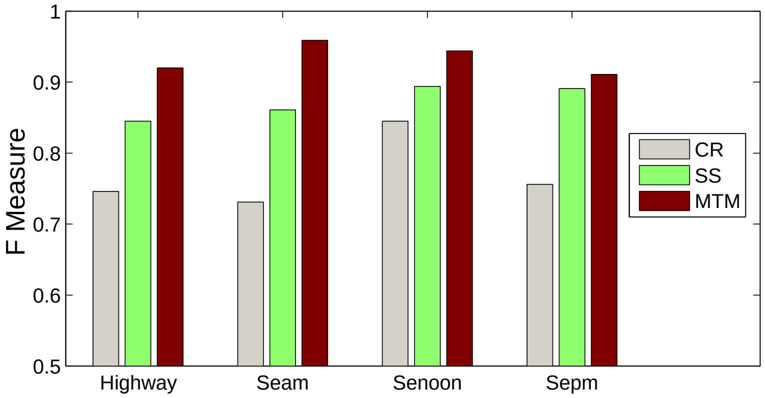

Figure 17: Foreground Detection results on sequences from the SZTAKI benchmark Set. Result shown are for the Constant Ratio (CR), Statistical Shadow (SS), and the proposed MTM methods. Adapted from [8].

Further details of the shadow detection approach using MTM can be found in [8].

7 Conclusions

In this paper, a fast pattern matching scheme called *Matching by Tone Mapping (MTM)* was introduced. The distance measure used is expressed as a minimization problem over all possible tone mappings. Thus, by definition, the MTM is invariant to non-linear tone mappings (both monotonic and non-monotonic). Furthermore, the MTM is shown to be a generalization of the NCC for non-linear mappings and actually reduces to NCC when mappings are restricted to be linear [30]. An efficient computation of the MTM is proposed requiring computation time similar to the fast implementation of NCC.

As mentioned above the MTM and MI approaches are similar in spirit. While MI maximizes the entropy reduction in $\mathbf{w}$ given $\mathbf{p}$, MTM maximizes the variance reduction in $\mathbf{w}$ given $\mathbf{p}$. The entropy and variance are two different measures of uncertainty. While variance is a quantitative measure preferring a compact distribution of samples, the entropy is a qualitative measure disregarding bin rearrangements. The use of variance rather than entropy is critical when a small number of samples are available. Nevertheless, although MTM demonstrates superior performance with respect to run time and stability under sparse samples, it relies on functional dependency between $\mathbf{p}$ and $\mathbf{w}$. When this assumption is violated, e.g. in multi-modal images (between which functional mapping does not necessarily exist), MI often outperforms MTM, although at the expense of longer running time. The following table summarizes the comparison between the MI and the MTM schemes.

	MI	MTM
Maximizes	entropy reduction	variance reduction
Speed	slow	fast
Bin size	sensitive	insensitive
Measure	qualitative	quantitative

References

[1] E. Arbel and H. Hel-Or. Shadow removal using intensity surfaces and texture anchor points. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(6):1202–1216, 2011.

[2] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. In *SIGGRAPH*, 2009.

[3] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *The European Conference on Computer Vision*, pages 404–417, 2006.

[4] C. Benedek and T. Sziranyi. Shadow detection in digital images and videos. In R. Lukac, editor, *Computational Photography: Methods and Applications*, pages 283–308. CRC Press, Boca Raton, FL, USA, 2010.

[5] J. Bigun and G. Granlund. Optimal orientation detection of linear symmetry. In *1st IEEE Int. Conf. on Computer Vision*, pages 433–438, 1987.

[6] R. Brunelli. *Template Matching Techniques in Computer Vision: Theory and Practice.* Wiley, 2009.

[7] A. Buades and B. Coll. A non-local algorithm for image denoising. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 60–65, 2005.

[8] E. Bullkich, I. Ilan, Y. Moshe, Y. Hel-Or, and H. Hel-Or. Moving shadow detection by nonlinear tone-mapping. In *Int. Conf. on Systems, Signals and Image Processing (IWSSIP)*, 2012.

[9] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. *The European Conference on Computer Vision*, pages 778–792, 2010.

[10] S.D. Conte and C. DeBoor. *Elementary Numerical Analysis.* McGraw-Hill, New York, 1972.

[11] G.S. Cox. Review: Template matching and measures of match in image processing. *University of Cape Town, TR*, 1995.

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, 2005.

[13] R.O. Duda, P.E. Hart, and D.G. Stork. Pattern classification. *John Willey & Sons*, 2001.

[14] R.C. Gonzalez and R.E. Woods. *Digital Image Processing.* Prentice-Hall, 2006.

[15] C.H. Goulden. *Methods Of Statistical Analysis.* John Wiley and Sons, 1939.

[16] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50, 1988.

[17] H. Hel-Or. Matlab code "http://www.faculty.idc.ac.il/toky/software/software.htm".

[18] Y. Hel-Or and D. Shaked. A discriminative approach for wavelet denoising. *IEEE Trans. on Image Processing*, 17(4):443–457, 2008.

[19] N. Jojic, B. J. Frey, and A. Kannan. Epitomic analysis of appearance and shape. In *IEEE Int. Conference on Computer Vision*, volume 1, pages 34–41, 2003.

[20] J.P. Lewis. Fast normalized cross-correlation. In *Vision Interface*, pages 120–123, 1995.

[21] S. Kagarlitsky, Y. Moses, and Y. Hel-Or. Piecewise-consistent color mappings of images acquired under various conditions. In *The 12th IEEE International Conference on Computer Vision*, Kyoto, Japan, Sept 2009.

[22] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[23] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[24] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):971–987, 2002.

[25] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Trans. on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[26] M. Piccardi. Background subtraction techniques: A review. In *IEEE SMC Intl. Conf. on Systems, Man and Cybernetics*, 2004.

[27] F. Porikli. Integral histogram: a fast way to extract histograms in cartesian spaces. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 829–836, 2005.

[28] A. Prati, I. Mikic, M.M. Trivedi, and R. Cucchiara. Detecting moving shadows: Algorithms and evaluation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(7):918–923, 2003.

[29] A. Roche, G. Malandain, X. Pennec, and N. Ayache. The correlation ratio as a new similarity measure for multimodal image registration. *Lecture Notes in Computer Science*, 1496:1115–1124, 1998.

[30] S. Ross. *A First Course in Probability.* Prentice Hall, fifth edition, 1998.

[31] A. Sibiryakov. Statistical template matching under geometric transformations. In *Int. conf. on Discrete geometry for computer imagery*, pages 225–237, 2008.

[32] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani. Summarizing visual data using bidirectional similarity. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[33] J. Stander, R. Mech, and J. Ostermann. Detection of moving cast shadows for object segmentation. *IEEE Trans. Multimedia*, 1(1):65–76, 1999.

[34] T. Hastie and R. Tibshirani and J.H. Friedman. *The Elements of Statistical Learning.* Springer, 2003.

[35] P. Viola and W.M. Wells. Alignment by maximization of mutual information. *International journal of computer vision*, 24(2):137–154, 1977.

[36] Y. Wei and L. Tao. Efficient histogram-based sliding window. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3003–3010, 2010.

[37] M. Zibulevski. Code at "http://ie.technion.ac.il/∼mcib".

Appendix A

We prove that $D(\mathbf{p}, \mathbf{w})$ measures the ratio between the conditional variance of $\mathbf{w}|\mathbf{p}$ and the variance of $\mathbf{w}$. The pattern $\mathbf{p}$ and the candidate window $\mathbf{w}$ are considered as random variables with m samples p_i and w_i, $i = 1..m$, respectively.

Theorem 1.

$$D(\mathbf{p}, \mathbf{w}) = \frac{E\left[\mathrm{var}(\mathbf{w}|\mathbf{p})\right]}{\mathrm{var}(\mathbf{w})}$$

where $E[\cdot]$ *is taken over all sample values* p_i *of* $\mathbf{p}$.

Proof 1.
Let $\mathbf{p}^j$ be the j-th slice of $\mathbf{p}$. All the pattern slices are mutually exclusive, thus $\sum_j \mathbf{p}^j = \mathbf{1}$. Consequently, d_1 can be rewritten as:

$$\begin{aligned} d_1 &= \sum_j \mathbf{p}^j \cdot (\mathbf{w} \odot \mathbf{w}) - \sum_j \frac{(\mathbf{p}^j \cdot \mathbf{w})^2}{|\mathbf{p}^j|} = \\ &= \sum_j |\mathbf{p}^j| \frac{\mathbf{p}^j \cdot (\mathbf{w} \odot \mathbf{w})}{|\mathbf{p}^j|} - \sum_j |\mathbf{p}^j| \left(\frac{\mathbf{p}^j \cdot \mathbf{w}}{|\mathbf{p}^j|} \right)^2 = \\ &= \sum_j |\mathbf{p}^j| \left(\frac{\mathbf{p}^j \cdot (\mathbf{w} \odot \mathbf{w})}{|\mathbf{p}^j|} - \left(\frac{\mathbf{p}^j \cdot \mathbf{w}}{|\mathbf{p}^j|} \right)^2 \right) \end{aligned}$$

Let $g_{\mathbf{p}}^j$ denote the set of indexes of non-zero entries of slice $\mathbf{p}^j$. Therefore,

$$\mathbf{p}^j \cdot \mathbf{w} = \sum_{i \in g_{\mathbf{p}}^j} \mathbf{w}_i, \quad \text{and as such,} \quad \frac{\mathbf{p}^j \cdot \mathbf{w}}{|\mathbf{p}^j|} = E[\mathbf{w}|\mathbf{p}^j]$$

Therefore,

$$\begin{aligned} d_1 &= \sum_j |\mathbf{p}^j| \left(\frac{\mathbf{p}^j \cdot (\mathbf{w} \odot \mathbf{w})}{|\mathbf{p}^j|} - \left(\frac{\mathbf{p}^j \cdot \mathbf{w}}{|\mathbf{p}^j|} \right)^2 \right) = \\ &= \sum_j |\mathbf{p}^j| \left(E\left[\mathbf{w} \odot \mathbf{w} \mid \mathbf{p}^j\right] - E^2\left[\mathbf{w} \mid \mathbf{p}^j\right] \right) \\ &= \sum_j |\mathbf{p}^j| \ \mathrm{var}\left(\mathbf{w} \mid \mathbf{p}^j\right) = mE\left[\mathrm{var}\left(\mathbf{w} \mid \mathbf{p}\right)\right] \end{aligned} \tag{23}$$

where the expectation in the last equation is taken over all $\{\mathbf{p}^j\}$. From Equation 23 it follows that

$$D(\mathbf{p}, \mathbf{w}) = \frac{d_1}{d_2} = \frac{E\left[\mathrm{var}\left(\mathbf{w} \mid \mathbf{p}\right)\right]}{\mathrm{var}\left(\mathbf{w}\right)} \qquad \square$$

Appendix B

MTM is related to the Fisher Linear Discriminant [13], in which the goal is to maximize inter-class variance while minimizing intra-class variance. From Theorem 2 we have that minimizing $D(\mathbf{p},\mathbf{w})$ is equal to maximizing $\tilde{D}(\mathbf{p},\mathbf{w})$. Using the law of total variance [30]:

$$\begin{aligned}
\arg\max_{\mathbf{w}} \tilde{D}(\mathbf{p},\mathbf{w}) &= \arg\max_{\mathbf{w}} \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{\mathrm{var}(\mathbf{w})} && (24)\\
&= \arg\min_{\mathbf{w}} \frac{\mathrm{var}(\mathbf{w})}{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])} && (25)\\
&= \arg\min_{\mathbf{w}} \frac{E[\mathrm{var}(\mathbf{w}|\mathbf{p})] + \mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])} && (26)\\
&= \arg\min_{\mathbf{w}} (\frac{E[\mathrm{var}(\mathbf{w}|\mathbf{p})]}{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])} + 1) && (27)\\
&= \arg\min_{\mathbf{w}} \frac{E[\mathrm{var}(\mathbf{w}|\mathbf{p})]}{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])} && (28)\\
&= \arg\max_{\mathbf{w}} \frac{\mathrm{var}(E[\mathbf{w}|\mathbf{p}])}{E[\mathrm{var}(\mathbf{w}|\mathbf{p})]} && (29)\\
& && (30)
\end{aligned}$$

Considering the grayscale bins associated with $\mathbf{p}$ as classes, we have that minimizing $D(\mathbf{p},\mathbf{w})$ is similar to maximizing inter-class variance while minimizing intra-class variance.

Appendix C

When constraining tone mappings to be linear, pattern matching using MTM results in the NCC solution. Recall the original definition of MTM (Equation 1):

$$D(\mathbf{p},\mathbf{w}) = \min_{\mathcal{M}} \left\{ \frac{(\| \mathcal{M}(\mathbf{p}) - \mathbf{w}) \|^2}{m \cdot \mathrm{var}(\mathbf{w})} \right\} \quad (31)$$

Since the mappings are restricted to linear mappings: $\mathcal{M}(\mathbf{p}) = a\mathbf{p} + b$, we seek parameters a, b satisfying:

$$\min_{a,b} \|a\mathbf{p} + b - \mathbf{w}\|^2$$

It has been shown (e.g. [30] Ch. 7) that minimizing the above term gives:

$$\begin{aligned} a &= \frac{\mathrm{cov}(\mathbf{p},\mathbf{w})}{\mathrm{var}(\mathbf{p})} = \rho(\mathbf{p},\mathbf{w})\sqrt{\frac{\mathrm{var}(\mathbf{w})}{\mathrm{var}(\mathbf{p})}} && (32) \\ b &= E[\mathbf{w}] - aE[\mathbf{p}] = E[\mathbf{w}] - \rho(\mathbf{p},\mathbf{w})E[\mathbf{p}]\sqrt{\frac{\mathrm{var}(\mathbf{w})}{\mathrm{var}(\mathbf{p})}} && (33) \end{aligned}$$

where $\rho(\mathbf{p},\mathbf{w}) = \frac{\mathrm{cov}(\mathbf{p},\mathbf{w})}{\sqrt{\mathrm{var}(\mathbf{p})\mathrm{var}(\mathbf{w})}}$. Substituting a and b into Equation 31 we obtain [30]:

$$D(\mathbf{p},\mathbf{w}) = 1 - \rho^2(\mathbf{p},\mathbf{w})$$

and thus,

$$\tilde{D}(\mathbf{p},\mathbf{w}) = \rho^2(\mathbf{p},\mathbf{w}) \qquad \square$$

Appendix D

We show that $Q(\mathbf{p})$ is a tridiagonal matrix. Recall that $Q(\mathbf{p})$ is composed of column vectors:

$$Q(\mathbf{p}) = [\bar{\mathbf{p}}^1, \bar{\mathbf{p}}^2, \cdots, \bar{\mathbf{p}}^{k+1}]$$

where

$$\bar{\mathbf{p}}^j = \mathbf{p}^j - \tilde{\mathbf{p}}^j + \tilde{\mathbf{p}}^{j-1}$$

To simplify notations, we denote $Q(\mathbf{p})$ as Q. The matrix $G = Q^T Q$ then reads:

$$[G(\mathbf{p})](i,j) = \bar{\mathbf{p}}^i \cdot \bar{\mathbf{p}}^j = (\mathbf{p}^i - \tilde{\mathbf{p}}^i + \tilde{\mathbf{p}}^{i-1}) \cdot (\mathbf{p}^j - \tilde{\mathbf{p}}^j + \tilde{\mathbf{p}}^{j-1}) \tag{34}$$

Since the two sets of vectors $\{\mathbf{p}^i\}$ and $\{\tilde{\mathbf{p}}^i\}$ are each mutually orthogonal, we obtain that the above term vanishes for all indices (i,j) except for indices satisfying $|i-j| \leq 1$. This results in a tridiagonal matrix:

$$G(\mathbf{p}) = \begin{bmatrix} \varphi_p^1 & \psi_p^1 & & & 0 \\ \psi_p^1 & \varphi_p^2 & \psi_p^2 & & \\ & \psi_p^2 & \varphi_p^3 & \ddots & \\ & & \ddots & \ddots & \psi_p^k \\ 0 & & & \psi_p^k & \varphi_p^{k+1} \end{bmatrix} \tag{35}$$

where for the main diagonal we have:

$$\varphi_p^j = \bar{\mathbf{p}}^j \cdot \bar{\mathbf{p}}^j = \sum_{i \in g_{\mathbf{p}}^j} (1 - r(p_i))^2 + \sum_{i \in g_{\mathbf{p}}^{j-1}} (r(p_i))^2, \qquad \text{for } j = 1 \cdots k+1$$

and for the off diagonal entries:

$$\psi_p^j = \bar{\mathbf{p}}^j \cdot \bar{\mathbf{p}}^{j+1} = \sum_{i \in g_{\mathbf{p}}^j} (1 - r(p_i))r(p_i), \qquad \text{for } j = 1 \cdots k$$

Note, that in the summation above we assume that $g_{\mathbf{p}}^0 = g_{\mathbf{p}}^{k+1} = \emptyset$

Appendix E

In this section we solve for $\hat{\beta}$ in the system $G\hat{\beta} = Q^T\mathbf{w}$ where G is a symmetric tridiagonal matrix, $Q = Q(\mathbf{p})$ is the SLT-PWL matrix of $\mathbf{p}$ and $\rho_j = \mathbf{w} \cdot \bar{\mathbf{p}}^j$:

$$\underbrace{\begin{bmatrix} \varphi_p^1 & \psi_p^1 & & & 0 \\ \psi_p^1 & \varphi_p^2 & \psi_p^2 & & \\ & \psi_p^2 & \varphi_p^3 & \ddots & \\ & & \ddots & \ddots & \psi_p^k \\ 0 & & & \psi_p^k & \varphi_p^{k+1} \end{bmatrix}}_{G(\mathbf{p})=Q^TQ} \cdot \underbrace{\begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \\ \vdots \\ \hat{\beta}_{k+1} \end{bmatrix}}_{\hat{\beta}} = \underbrace{\begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \vdots \\ \rho_{k+1} \end{bmatrix}}_{Q(\mathbf{p})^T\mathbf{w}} \tag{36}$$

Since G is tridiagonal, the above linear system can be solved with a linear number of operations using a simplified version of the Gaussian elimination method [10]. The process involves a forward sweep that eliminates the ψ_p^i's below the main diagonal, followed by a backward substitution that produces the solution.

In the first step the above system is modified to a new set of equations using Gaussian elimination:

$$\begin{bmatrix} 1 & \omega_p^1 & & & 0 \\ 0 & 1 & \omega_p^2 & & \\ & 0 & 1 & \ddots & \\ & & \ddots & \ddots & \omega_p^k \\ & & & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \\ \vdots \\ \hat{\beta}_{k+1} \end{bmatrix} = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_{k+1} \end{bmatrix} \tag{37}$$

where the new coefficients are calculated as follows:

$$\omega_p^i = \begin{cases} \dfrac{\psi_p^1}{\varphi_p^1} & for \quad i = 1 \\ \dfrac{\psi_p^i}{\varphi_p^i - \omega_p^{i-1}\psi_p^{i-1}} & for \quad i = 2, 3, \ldots, k \end{cases} \tag{38}$$

and

$$\sigma_i = \begin{cases} \dfrac{\rho_i}{\varphi_p^1} & for \quad i = 1 \\ \\ \dfrac{\rho_i - \sigma_{i-1}\psi_p^{i-1}}{\varphi_p^i - \omega_p^{i-1}\psi_p^{i-1}} & for \quad i = 2, 3, \ldots, k+1 \end{cases} \tag{39}$$

The solution is then obtained using backward substitution:

$$\begin{aligned} \hat{\beta}_{k+1} &= \sigma_{k+1} \\ \hat{\beta}_i &= \sigma_i - \omega_p^i \, \hat{\beta}_{i+1}, \quad \text{for } i = k, k-1, \cdots, 1 \end{aligned} \tag{40}$$

Note, that during the elimination step the coefficients $\{\omega_i\}$ are calculated only once for all candidate windows, while $\{\sigma_i\}$ and $\hat{\beta}$ must be calculated for each window. Since $Q^T\mathbf{w}$ is calculated using $O(m)$ operations (Section 4) and calculating $\hat{\beta}$ requires an additional $O(k)$ operations the entire process requires $O(m+k)$ operations.