A PRELIMINARY CASE FOR HIRSCHMAN TRANSFORM VIDEO CODING

Joseph P. Havlicek[1], *Tristan N. Arian*[1], *Hoda Soltani*[2], *Tomasz Przebinda*[3], *and Murad Özaydın*[3]

[1]School of Electrical and Computer Engineering, University of Oklahoma, USA

[2]Center for Intelligent Transportation Systems, University of Oklahoma, USA

[3]Department of Mathematics, University of Oklahoma, USA

ABSTRACT

The use of Hirschman transforms for video coding has not been considered previously, primarily due to the lack of any previous construction of a real-valued Hirschman transform. In this paper, we introduce a new 2D separable 8×8 Hirschman transform that, for the first time, is inherently real and integer-valued. We apply this new transform to perform block-based video coding. Our preliminary results on the well-known *Foreman* sequence show significantly superior coefficient quantization performance relative to both the 2D DCT and the H.264 8×8 integer transform. We also study the energy compaction characteristics of the new transform. Preliminary results indicate that, in contrast to the DCT, the Hirschman transform coefficients tend to cluster around a few integer values on average, suggesting that m-ary run-length coding may provide a particularly efficient encoding after quantization of the Hirschman transform coefficients.

Index Terms— video compression, quantization, DCT, Hirschman transform, H.264

1. INTRODUCTION

The discrete cosine transform (DCT) and integer approximations thereto have reigned as the block transforms of choice for ubiquitous video coding standards ranging from H.261 to H.265/HEVC and beyond [1–7]. These coding methods generally achieve compression through a sequence of three main algorithmic steps, including:

1. inter or intra prediction of pixel values (*lossless entropy reduction*),
2. application of a block-based transform followed by quantization of the transform coefficients (*lossy bit rate reduction*), and
3. run-length and entropy coding of the quantized transform coefficients (*lossless bit rate reduction*).

The arguments supporting use of the DCT for the block transform step derive chiefly from [8]. Specifically, the DCT tends to achieve decorrelation or "energy compaction" by concentrating most of the energy that is present in a block of pixels into a few (typically low frequency) transform coefficients. Reordering the quantized transform coefficients into a 1D array using, e.g., the well-known "zigzag ordering" [2, p. 55] then tends to produce a sparse string of nonzero quantized transform coefficients interspersed with runs of zero values that can be efficiently run-length coded.

The use of multiple transforms in a competition scheme has also been investigated recently; see [9] and the references therein for a good overview. However, 2D Hirschman transforms have been considered previously for image analysis and enhancement only in [10–12], all of which used complex-valued Hirschman transforms and, to the best of our knowledge, Hirschman transform video coding has not been previously considered.

In this paper, we introduce a new 2D separable 8×8 discrete Hirschman transform that, for the first time, is inherently real. Moreover, for integer-valued input signals such as typical consumer video, the new transform is also inherently integer-valued. These two properties lead to several desirable characteristics including extremely efficient computation and excellent quantization performance. While limited in scope, our preliminary experiments on the luminance component of the well-known *Foreman* video test sequence demonstrate significantly superior PSNR performance relative to the 8×8 DCT and the H.264 8×8 approximate DCT integer transform [2, 13].

2. HIRSCHMAN TRANSFORMS

Consider the ring $A = \mathbb{Z}/N\mathbb{Z}$ with addition and multiplication modulo N. Let $x \in \ell^2(A)$ so that $x : [0,\ N-1] \to \mathbb{C}$. The discrete Fourier transform (DFT) of x is given by

$$(\mathcal{F}x)[k] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] W_N^{nk}, \tag{1}$$

where $W_N = \exp(-j2\pi/N)$ and j denotes the imaginary unit. Let $u \in \ell^2(A)$ such that $||u||_{\ell^2} = 1$. As in [14–17], we quantify the uncertainty or localization of u by the entropy

$$H(u) = -\sum_{n=0}^{N-1} |u[n]|^2 \ln\left(|u[n]|^2\right) \tag{2}$$

and the joint time-frequency localization of u by

$$H_p(u) = pH(u) + (1-p)H(\mathcal{F}u). \tag{3}$$

Orthogonal transforms constructed using a basis for $\ell^2(A)$ wherein each basis signal u admits optimal joint localization in the sense of minimizing $H_p(u)\ \forall\ 0 \le p \le 1$ are called *Hirschman optimal transforms* (HOT's). Techniques for constructing HOT bases with N a perfect square were demonstrated in [16, 17], but are unknown for general N.

Techniques for constructing bases that minimize $H_p(u)$ for individual $p \neq \frac{1}{2}$ are also unknown. However, orthonormal bases that minimize $H_{\frac{1}{2}}(u)$ may be constructed as in [11, 16, 17]. Transforms on $\ell^2(A)$ that utilize such bases are called *Hirschman transforms*.

3. REAL 8×8 BLOCK HIRSCHMAN TRANSFORM

Let B be an additive subgroup of A and let $\mathbb{I}_B(n)$ be the indicator function of B. Then an orthonormal basis for $\ell^2(A)$ wherein every basis signal minimizes $H_{\frac{1}{2}}$ may be generated by applying compositions of the following operators to the signal $v(n) = \mathbb{I}_B(n)/||\mathbb{I}_B||_{\ell^2}$ [16, 17]:

1. translation modulo N: $T_D v[n] = v[(n - D) \mod N]$,
2. modulation: $M_a v[n] = W_N^{-an} v[n]$.

Upon choosing $N = 8$, $B = \{0,\ 4\}$ and $a = 1$, we obtain

$$v[n] = \frac{1}{\sqrt{2}}[1\ 0\ 0\ 0\ 1\ 0\ 0\ 0]^T \tag{4}$$

and the basis signals are given by the rows (equivalently, the columns) of the involutary matrix

$$\mathcal{B} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}. \tag{5}$$

This defines a 1D Hirschman transform on $\ell^2(\mathbb{Z}/8\mathbb{Z})$ given by

$$y = \mathcal{B}x, \tag{6}$$
$$x = \mathcal{B}^{-1}y = \mathcal{B}y. \tag{7}$$

For 8×8 image blocks $\mathbf{X}$, a separable 2D Hirschman transform may then be formulated according to

$$\mathbf{Y} = \mathcal{B}\mathbf{X}\mathcal{B}^T = \mathcal{B}\mathbf{X}\mathcal{B}, \tag{8}$$
$$\mathbf{X} = \mathcal{B}^{-1}\mathbf{Y}\left(\mathcal{B}^T\right)^{-1} = \mathcal{B}\mathbf{Y}\mathcal{B}. \tag{9}$$

Now let $\mathcal{B}_2 = \sqrt{2}\mathcal{B}$. We define a second separable 8×8 Hirschman transform $\mathcal{H}_2$ according to

$$\mathbf{Y} = \mathcal{H}_2\mathbf{X} = \mathcal{B}_2\mathbf{X}\mathcal{B}_2, \tag{10}$$
$$\mathbf{X} = \mathcal{H}_2^{-1}\mathbf{Y} = \left(\mathcal{B}_2\mathbf{Y}\mathcal{B}_2\right)/4. \tag{11}$$

When the image block $\mathbf{X}$ is real and integer-valued, as is typical in practical video applications, this $\mathcal{H}_2$ transform admits several highly desirable numerical properties including:

- the $\mathcal{H}_2$ transform coefficients $\mathbf{Y}$ in (10) are also *real* and *integer-valued*;
- computation of the forward transform (10) requires additions only;
- computation of the inverse transform (11) requires only additions and bit shift operations.

When the pixels of the image block $\mathbf{X}$ are in the range $0 \leq x_{i,j} \leq 255$, which is typical, the $\mathcal{H}_2$ transform coefficients $\mathbf{Y}$ are in the range $-510 \leq y_{i,j} \leq 1{,}020$. If the $\mathcal{H}_2$ transform is instead applied to a block of intra or inter prediction residuals (as in, e.g., H.264 and HEVC), we obtain $-1{,}020 \leq y_{i,j} \leq 1{,}020$. In either case (application to an image block or to a block of prediction residuals), the $\mathcal{H}_2$ transform provides perfect inversion at a signed integer precision of only 11 bits for the transform coefficients. For video coding, this provides a *significant* advantage relative to both the 2D DCT and the H.264 8×8 integer transform process [2, 3, 13].

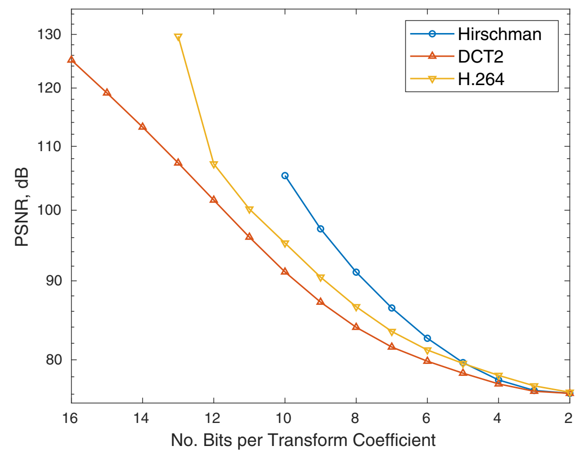

Fig. 1. PSNR as a function of N_{bits}, the number of bits used for representation of each signed integer transform coefficient. For the $\mathcal{H}_2$ transform, MSE=0 and PSNR=∞ for $N_{\text{bits}} \geq 11$. For the 8×8 H.264 transform, MSE=0 and PSNR=∞ for $N_{\text{bits}} \geq 14$.

4. QUANTIZATION PERFORMANCE EXAMPLES

We illustrate the coefficient quantization performance of the new 8×8 Hirschman transform $\mathcal{H}_2$ developed in Section 3 by running experiments on the luminance component of the well-known *Foreman* video test sequence at CIF resolution. The number of frames is 300. Motion estimation (ME) and motion compensation (MC) were performed from the previous frame at a block size of 8×8 using a search range of ±3 × 3 pixels at a resolution of one pixel to compute 8×8 inter-prediction residual blocks for frames 2 through 300. Note that, for these preliminary experiments, MC and ME were performed using the *original* pixel values from the preceding frame, not the decoded values. Thus, frame-to-frame decoding errors were not considered in these experiments.

We applied the $\mathcal{H}_2$ transform to the residual blocks. For comparison, we also applied the 2D DCT (using the Matlab `dct2` implementation) and the H.264 8×8 integer transform process (implemented as described in [2, Ch. 7]).

Fig. 1 shows PSNR averaged over all blocks of all frames as a function of N_{bits}, the number of bits used to represent each transform coefficient, for all three transforms considered. After floating point transformation, all 64 coefficients were truncated to a binary representation of $16 \geq N_{\text{bits}} \geq 2$ bits with bit used for sign and $N_{\text{bits}} - 1$ bits used for magnitude. The new $\mathcal{H}_2$ transform developed in Section 3 delivers a PSNR advantage relative to the other two transforms for $N_{\text{bits}} \geq 5$ and produces MSE=0/PSNR=∞ for $N_{\text{bits}} \geq 11$.

With regards to the H.264 integer transform, it should be noted that this experiment involves a range of equivalent QP values that is *much* larger than the range actually allowed by the H.264 standard [13]. QP can be back-calculated from the quantization step size Q_{step} as described in [2, Sec. 7.2.3.5]. For pixel values in the range $[0, 255]$ the residuals are in the range $[-255, 255]$. As a base case we consider QP = 4 (which gives Q_{step} = 1) and the H.264 8×8 integer transform coefficients are in the range $[-2{,}040, 2{,}040]$, thus requiring an N_{bits} = 12-bit binary representation. For N_{bits} = 11, we obtain Q_{step} = 2 and QP = 10; for N_{bits} = 10, we have Q_{step} = 4 and QP = 16, etc.

Fig. 2 shows PSNR averaged over all blocks of all frames as

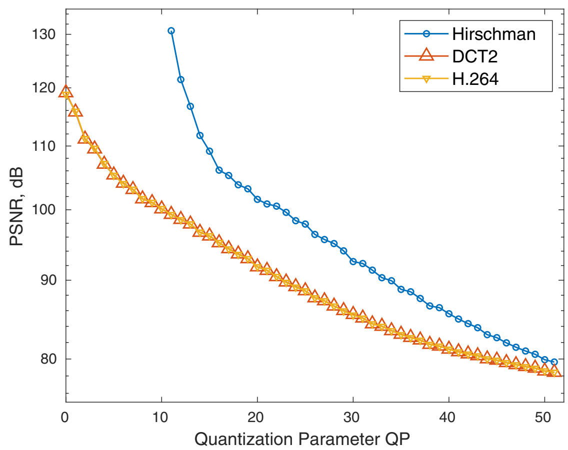

Fig. 2. PSNR as a function of quantization parameter QP for QP $\in [0, 51]$. DCT and H.264 are almost visually indistinguishable at this scale. For $\mathcal{H}_2$, PSNR=∞ when QP ≤ 10.

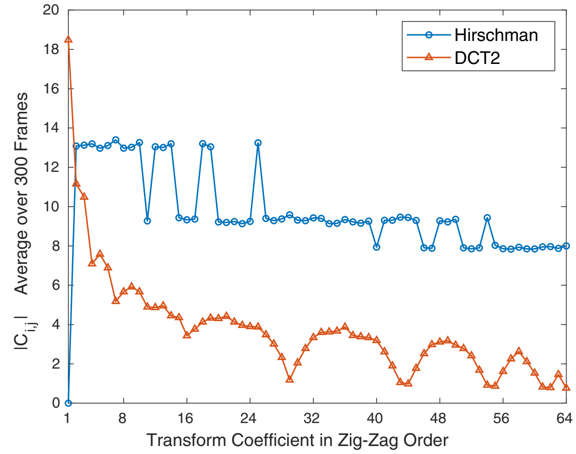

Fig. 3. Magnitudes of 64 transform coefficients, averaged over all blocks and frames, displayed in zig-zag order. DCT shows decay as expected. $\mathcal{H}_2$ shows clustering around a few integer attractors.

a function of the quantization parameter QP. Calculations were performed as in [2, Sec. 7.2.3.9], which is quite different from the calculations that led to Fig. 1. In particular, for $\mathcal{H}_2$ and DCT the forward transforms were computed, divided by Q_{step} using floating point arithmetic, and then rounded. The rounded coefficients were then multiplied by Q_{step} and inverted using floating point arithmetic. Finally, the inverted pixel values were rounded to the nearest integer. All of these calculations were performed at full floating point precision without regard to the number of bits required. Also note: since the H.264 8×8 integer transform and the 2D DCT both range from $-2{,}040$ to 2,040 at $Q_{\text{step}} = 1$ (QP = 4), whereas the $\mathcal{H}_2$ transform only ranges from $-1{,}020$ to 1,020, the $\mathcal{H}_2$ forward transform was scaled by 2 and the $\mathcal{H}_2^{-1}$ reverse transform was scaled by 1/2 for a fair comparison of PSNR with regards to quantization by QP.[1]

As we pointed out in Section 1, the 8×8 DCT coefficients tend to decay towards zero rather rapidly. This is shown in Fig. 3 for the *Foreman* sequence luminance component residuals, where the average magnitudes of the 64 DCT coefficients (averaged over all blocks of all frames) are shown in zig-zag order [2, 13]. This supports the idea that sufficient quantization of the DCT coefficients (or H.264 integer transform coefficients) should tend to produce a sparse string of nonzero quantized coefficients interspersed with relatively long runs of zero values that may be efficiently run-length coded.

By contrast, the $\mathcal{H}_2$ transform coefficients do not decay towards zero, as also shown in Fig. 3. This occurs because the Hirschman transform basis functions minimize the joint uncertainty measure $H_{\frac{1}{2}}$ which weights localization in the space and frequency domains equally, whereas the space and frequency domain localization of the DCT basis functions are decidedly *unequal*. However, Fig. 3 also shows that, on average at least, the $\mathcal{H}_2$ transform coefficients tend to cluster around just a few integer values. This strongly suggests that the quantized $\mathcal{H}_2$ transform coefficients of individual residual blocks may be amenable to extremely efficient m-ary run-length encoding – perhaps even more efficient than the zero-value run-length coding applied to the quantized DCT coefficients by existing video coding standards. While this is an idea that certainly deserves to be fully developed and evaluated, doing so is beyond the scope of this short, preliminary paper.

Finally, to translate the results shown in Figs. 1-3 into a more concrete notion of visual image quality, Fig. 4 shows actual decoded luminance component video frames from the experiment of Fig. 2 for the $\mathcal{H}_2$ and H.264 8×8 integer transforms. These results are for frame 181, which is the frame that produced the largest MSE in the experiment of Fig. 2. As was typical for most frames of this sequence, the decoding errors are virtually invisible for low to moderate QP values (indeed, for some frames the errors were hard to see even at QP = 51 – recall that the block transform and quantization processes were applied only to the inter-prediction residuals and that these residuals were free of frame-to-frame decoding error propagation in these experiments). However, as shown in the rightmost column of Fig. 4 for frame 181 among others the visual differences between the $\mathcal{H}_2$ and H.264 transforms at QP = 51 are quite dramatic.

5. DISCUSSION

We introduced a new 2D Hirschman block transform that is inherently both real and integer-valued. The new transform admits many desirable numerical properties that facilitate fast computation and perfect inversion at a coefficient bit depth of 11. We demonstrated superior quantization performance in terms of PSNR relative to the DCT and the H.264 8×8 integer transform. Important remaining future work includes determining how the results of Fig. 3 can be leveraged into efficient m-ary run-length coding, showing the performance gain when the new transform is fully integrated into the test models of existing video coding standards, and formulating related separable and nonseparable Hirschman transforms for a variety of different block sizes.

6. REFERENCES

[1] A. M. Tekalp, *Digital Video Processing*, Prentice Hall PTR, Upper Saddle River, NJ, 1995.

[2] I. E. Richardson, *The H.264 Advanced Video Compression Standard*, Wiley, West Sussex, UK, 2nd edition, 2010.

[1] As a fine point, also note that, for the graph of Fig. 2 at QP=16, rounding the $\mathcal{H}_2$ transform coefficients produces a larger MSE than at QP=17,18,19. For this reason, the graph at QP=16 (and *only* at QP=16) shows the PSNR obtained by truncation of the $\mathcal{H}_2$ coefficients rather than rounding.

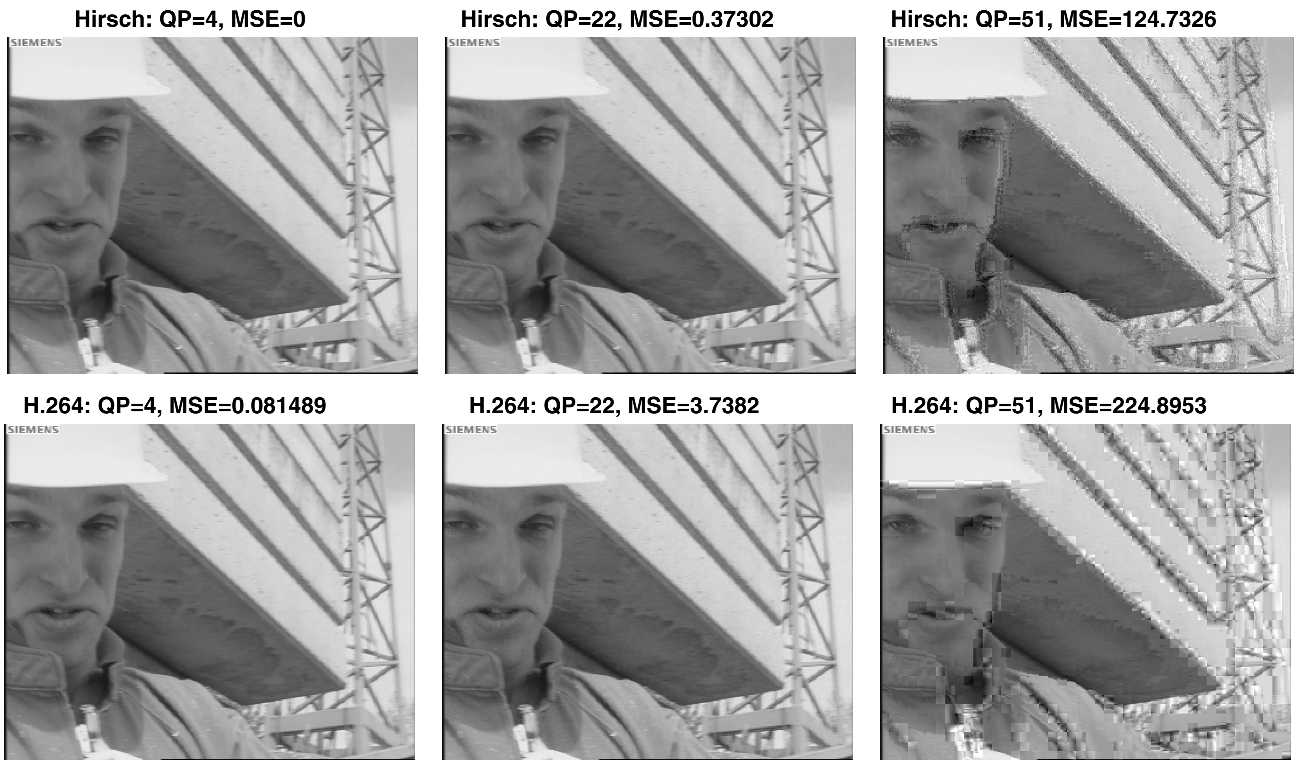

Fig. 4. Luminance reconstructions of *Foreman* frame 181 (the frame producing the worst MSE in our experiments) after transform, quantization, and inverse transform. The top row shows decoded results for the new $\mathcal{H}_2$ Hirschman transform developed in Section 3; the bottom row is for the H.264 8×8 integer transform process. The three columns correspond to QP = 4/22/51. MSE is shown above each test case. While $\mathcal{H}_2$ delivers superior MSE at all QP, the visual difference is dramatic at QP $= 51$.

[3] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[4] V. Sze, M. Budagavi, and G. J. Sullivan, Eds., *High Efficiency Video Coding (HEVC): Algorithms and Architectures*, Springer, 2014.

[5] M. Wien, *High Efficiency Video Coding: Coding Tools and Specification*, Springer, 2015.

[6] J. Bankoski, *et al.*, "Towards a next generation open-source video codec," in *Vis. Inf. Process. Commun. IV*, Feb. 2013, vol. 8666 of *Proc. SPIE*, pp. 866606–01–866606–13.

[7] Y. Chen, *et al.*, "An overview of core coding tools in the AV1 video codec," in *Proc. Picture Coding Symp.*, San Francisco, CA, Jun. 24-27, 2018, pp. 41–45.

[8] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan. 1974.

[9] T. Biatek, V. Lorcy, and P. Philippe, "Transform competition for temporal prediction in video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 815–826, Mar. 2019.

[10] W.-L. Hsue, S.-C. Pei, and J.-J. Ding, "Efficient discrete fractional Hirschman optimal transform and its application," in *Proc. IEEE Int'l. Conf. Acoust., Speech, Signal Process.*, Prague, Czech Republic, May 22-27, 2011, pp. 3604–3607.

[11] R. Thomas, V. DeBrunner, and L. DeBrunner, "Fixed-point implementation of discrete Hirschman transform," in *Proc. 52nd Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Oct. 28-31, 2018, pp. 1507–1511.

[12] B. Narsimha, E. Suresh, K. P. Chandar, and B. Komuraiah, "Enhancement of color images in HOT domain with quantitative measurements using entropy and relative entropy," in *Proc. Int'l. Conf. Signal, Image Process.*, Chennai, India, Dec. 15-17, 2010, pp. 34–38.

[13] "Advanced video coding for generic audiovisual services," *Recommendation ITU-T H.264 and International Standard ISO/IEC 14496-10*, Jun. 2019.

[14] V. DeBrunner, M. Özaydın, and T. Przebinda, "Resolution in time-Frequency," *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 783–788, Mar. 1999.

[15] V. DeBrunner, M. Özaydın, and T. Przebinda, "Analysis in a finite time-frequency plane," *IEEE Trans. Signal Process.*, vol. 48, no. 6, pp. 1831–1832, Jun. 2000.

[16] T. Przebinda, V. DeBrunner, and M. Özaydın, "The optimal transform for the discrete Hirschman uncertainty principle," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 2086–2090, Jul. 2001.

[17] V. DeBrunner, J. P. Havlicek, T. Przebinda, and M. Özaydın, "Entropy-based uncertainty measures for $L^2(\mathbb{R}^n)$, $\ell^2(\mathbb{Z})$, and $\ell^2(\mathbb{Z}/N\mathbb{Z})$ with a Hirschman optimal transform for $\ell^2(\mathbb{Z}/N\mathbb{Z})$," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2690–2699, Aug. 2005.