A Radial Transform for Video Compression

Aditya Somasundaram, Vikhyath Sai Kothamasu
Indian Institute of Technology Hyderabad, India.

***Abstract*—In this paper, a novel transformation inspired by the patterns in field dynamics is proposed. This transformation, coined as 'Radial Transform' has far-reaching effects and can capture a whole slew of transformations. The core idea behind the spherical construct of the Radial Transform is to exploit the rotational invariance present in signals in the transformed domain, unlike the conventional Discrete Cosine Transform (DCT) [1], which exploits the translational invariance of sinusoidal waves. The Radial Transform is employed within a video codec as a replacement for the DCT. The authors' results demonstrate that the Radial Transform achieves an increase in Peak Signal-to-Noise Ratio (PSNR) performance of 1.5 dB compared to DCT while maintaining the entropy in the regime around 0.7 and a minute difference in Multi-Scale Structural Similarity Index (MSSSIM) [2]. In texture analysis, superior compression results for texture-rich images and competitive performance in intricate patterns are observed. Incorporating the Radial Transform into the video codec presents a compelling proposition with advantages that significantly outweigh any associated disadvantages. This endeavor, initially inspired by field dynamics, has evolved into a novel and promising innovation.**

I. Introduction

Transformations, with their rich historical context, have played a pivotal role in various domains. They offer the means to represent an entity (signals) in different realms, facilitating its analysis. Fourier transforms [3], for example, reveal the frequency components within a signal. More recently, wavelet transforms [4] provide insights into the presence of specific frequencies and their temporal occurrences. It is a trade-off between representing the signal purely in the time domain and purely in the frequency domain. This paper introduces a novel transform, taking inspiration from the patterns occurring in field dynamics. The behavior of point charges in space significantly influences the configuration of electric field. Altering the positions or strengths of these point charges results in time varying changes in the field's spatial distribution. Consequently, it becomes imperative to explore the inverse problem: determining the configuration of point charges responsible for a given electric field shape. In a sense, given the field, we should be able to localize these *sources of disturbances*.

The primary objective of this paper is to devise a methodology for characterizing the field configuration by effectively capturing the attributes of inherent disturbances. In practical applications, this methodology can revolutionize video compression techniques. This paper fundamentally introduces a novel transform coined as the 'Radial Transform' (RT), capable of processing functions and mapping them as outcomes influenced by source-induced disturbances.

Visualising videos as moving fields: An image may be interpreted as a two-dimensional signal, where each pixel at the coordinates (i, j) carries a specific intensity, denoted as $I[i, j]$. This concept can be extended to envision an image as a $3D$ terrain, where each point is represented by $(i, j, I[i, j])$. In the context of video, a sequence of such images is encountered, akin to a time-varying $3D$ landscape. The essence of every video codec lies in capturing the dynamic transformation of this terrain as it evolves over time.

Drawing inspiration from the principles of physics, the terrain is viewed as a dynamic field in which movements are driven by perturbations originating from sources and sinks. Encoding the change in terrain by capturing the properties of said disturbances is the core idea behind the transform [5] and its application in video compression (H.26x [6], [7], HEIF, MPEG [8]).

II. The Radial Transform

Let $\mathcal{A} \subset \mathbb{R}^n$ is finite and $|\mathcal{A}| = k$. Let f, g be two functions defined on the same domain: $f, g : \mathcal{A} \to \mathbb{R}$. The domain is mapped to a set in a higher dimension using a bijective map Φ. We have a bijective map $\Phi : \mathcal{A} \to \mathcal{P}$, $\mathcal{P} \subset \mathbb{R}^{n+1}$. Another bijective map Ψ_f, which depends on the domain and co-domain of f is defined. $\Psi_f : \mathcal{A} \to \mathcal{Q}$, $\mathcal{Q} \subset \mathbb{R}^{n+1}$, $\Psi_f(\mathbf{x}) = \Psi(\mathbf{x}, f(\mathbf{x}))$, such that $\mathcal{P} \cap \mathcal{Q} = \emptyset$. The function Ψ is left to the designer for customization. Throughout the paper we will refer to $\boldsymbol{\alpha} \in \mathcal{P}$ as the *source points* and $\boldsymbol{\beta} \in \mathcal{Q}$ as the *target points*. Now that the set of source points and target points are in the same space, $\mathbb{R}^{n+1}$, a function depicting how a source point located at $\boldsymbol{\alpha}$ will affect a target point located at $\boldsymbol{\beta}$ is defined: $h_{\boldsymbol{\theta}} : \mathcal{P} \times \mathcal{Q} \to \mathbb{R}$, $\boldsymbol{\theta} \in \mathbb{R}^m$ (analogous to point charges acting on each other and the corresponding $h_{\boldsymbol{\theta}}(\boldsymbol{r_1}, \boldsymbol{r_2}) = \frac{kq_1q_2}{\|\boldsymbol{r_1} - \boldsymbol{r_2}\|^2}$). The definition of Radial Transform: for every $\mathbf{x} \in \mathcal{A}$,

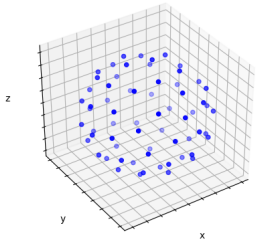
$$g(\mathbf{x}) = \sum_{\mathbf{x}' \in \mathcal{A}} S_{\mathbf{x}'} h_{\boldsymbol{\theta}}(\Phi(\mathbf{x}'), \Psi_f(\mathbf{x})) \tag{1}$$

where $S_{\mathbf{x}'} \in \mathbb{R}$. Let the *Strength* of point sources be denoted by $S_{\mathbf{x}'}$. We get k number of equations (i.e., for each $\mathbf{x} \in \mathcal{A}$, we get one equation). By solving the set of linear equations (of course assuming that the solution exists), we could get the values of strength associated with each point source. These values could vary depending on our selection of $h_{\boldsymbol{\theta}}(\Phi(\mathbf{x}'), \Psi_f(\mathbf{x}))$. Instead of representing a function with its values, we represent it using these sources. The strength of these sources could be considered to be the analysis coefficients of the function g with respect to f. Knowing f and the values of $S_{\mathbf{x}'}$, one can fully recover g. It should be interesting to investigate whether a unique choice h would lead to the vanishing of most $S_{\mathbf{x}'}$.

III. Spherical Construct of the Radial Transform

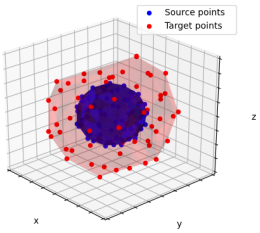
A. The construct

Going forward, the spherical construct of the Radial Transform is chosen. The radial symmetry [9]–[11] of the spherical



(a) The spherical even distribution of 64 points in $3D$.

(b) The 8×8 block which act as the target points.



(c) Example of an 8×8 block enveloped over the sphere.

Figure 1: Example visual of the spherical construct.

construct ties well with the radial symmetry we get from our $h_{\boldsymbol{\theta}}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ as defined in equation 2, and intuitively, this mapping should best exploit the redundancy during compression.

The functions f, g are mapped to the $n+1$ dimensional space using the following Φ and Ψ_f maps. $\mathcal{P}$ is chosen to consist of k points $\boldsymbol{\alpha_i}$ on the $n+1$ dimensional unit sphere such that they are distributed in a symmetric fashion [12]. An example for the same is shown in Fig. 1a. The functions Φ and Ψ_f are defined as $\Phi : \mathcal{A} \to \mathcal{P}$, such that points which are close in $\mathcal{A}$ are close in $\mathcal{P}$; $\Psi_f : \mathcal{A} \to \mathcal{Q}$, such that $\Psi_f(\mathbf{x}) = f(\mathbf{x}) \cdot \Phi(\mathbf{x})$. The point sources are considered to act like point charges. The factor (the effect a point source has on a unit charge in space) emanating will be inversely proportional to the distance between the point source entity and the unit entity. h is chosen in a similar fashion: For $\boldsymbol{\alpha} \in \mathcal{P}$, $\boldsymbol{\beta} \in \mathcal{Q}$ and $\theta \in \mathbb{R}$

$$h_\theta(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{\|\boldsymbol{\beta} - \boldsymbol{\alpha}\|^\theta} = \|\boldsymbol{\beta} - \boldsymbol{\alpha}\|^{-\theta} \tag{2}$$

Let $\boldsymbol{H}_\theta = \begin{bmatrix} \|\boldsymbol{\beta_1} - \boldsymbol{\alpha_1}\|^{-\theta} & \|\boldsymbol{\beta_1} - \boldsymbol{\alpha_2}\|^{-\theta} & \dots & \|\boldsymbol{\beta_1} - \boldsymbol{\alpha_k}\|^{-\theta} \\ \|\boldsymbol{\beta_2} - \boldsymbol{\alpha_1}\|^{-\theta} & \|\boldsymbol{\beta_2} - \boldsymbol{\alpha_2}\|^{-\theta} & \dots & \|\boldsymbol{\beta_2} - \boldsymbol{\alpha_k}\|^{-\theta} \\ \vdots & \vdots & \ddots & \vdots \\ \|\boldsymbol{\beta_k} - \boldsymbol{\alpha_1}\|^{-\theta} & \|\boldsymbol{\beta_k} - \boldsymbol{\alpha_2}\|^{-\theta} & \dots & \|\boldsymbol{\beta_k} - \boldsymbol{\alpha_k}\|^{-\theta} \end{bmatrix}$

The full matrix representation is as follows:

$$\boldsymbol{H}_\theta \times \begin{bmatrix} S_{\mathbf{x_1}} \\ S_{\mathbf{x_2}} \\ \vdots \\ S_{\mathbf{x_k}} \end{bmatrix} = \begin{bmatrix} g(\mathbf{x_1}) \\ g(\mathbf{x_2}) \\ \vdots \\ g(\mathbf{x_k}) \end{bmatrix} \tag{3}$$

Qualitatively, if the receiver has knowledge about f and $S_{\mathbf{x}'}$, a simple matrix multiplication would suffice to obtain g.

B. In the specific case of consecutive frames in videos

Consider two consecutive frames from a video, f_1, and f_2. The function f_1 maps each pixel coordinate in the first frame (represented as a 2D vector in $\mathbb{R}^2$) to its corresponding intensity value. For any pixel coordinate denoted by $\mathbf{x}$, $f_1(\mathbf{x})$ gives the intensity value at that pixel. The set $\mathcal{A}$ represents all the pixel coordinates in the frames. The bijective map Φ takes each pixel coordinate and maps it to a corresponding point on a unit sphere in $\mathbb{R}^3$. The map is a set of source points $\mathcal{P}$, where each point on the sphere represents and is represented by a pixel coordinate from the first frame and is equidistant from all its nearest neighbors (See Fig. 1c). The bijective map Ψ_{f_1} builds upon Φ by incorporating intensity information. Ψ_{f_1} takes a pixel coordinate $\mathbf{x}$ and its corresponding intensity $f_1(\mathbf{x})$, and maps them to a target point on a larger body in $\mathbb{R}^3$. This larger body, denoted by $\mathcal{Q}$, represents and is represented by all the pixel coordinates along with their corresponding intensity in the first frame (See Fig. 1c). The map Ψ_{f_1} positions points on the larger body based on combined information about location and intensity. Importantly, these two sets $\mathcal{P}$ and $\mathcal{Q}$ do not intersect. The function $h_{\boldsymbol{\theta}} : \mathcal{P} \times \mathcal{Q} \to \mathbb{R}$, modeled by a set of parameters $\boldsymbol{\theta}$ captures the influence that a source point (located on the first sphere) has on a target point (located on the larger body). The function $h_{\boldsymbol{\theta}}$ is left to the imagination of the designer. Lastly, the equation $f_2(\mathbf{x}) = \sum_{\mathbf{x}' \in \mathcal{A}} S_{\mathbf{x}'} h_{\boldsymbol{\theta}}(\Phi(\mathbf{x}'), \Psi_{f_1}(\mathbf{x}))$ should correspond to the second frame. This function takes a pixel coordinate $\mathbf{x}$ as input and provides an output value $f_2(\mathbf{x})$, which is the intensity of the second frame at $\mathbf{x}$. The value of $f_2(\mathbf{x})$ is a weighted sum of the influences from all the source points on the target point corresponding to $\mathbf{x}$ (as determined by $h_{\boldsymbol{\theta}}$). These weights are strengths, $S_{\mathbf{x}'}$, associated with each source point. By solving the system of equations for $f_2(\mathbf{x})$ for all pixel coordinates (which is possible due to the specific properties of the chosen maps and functions), we can effectively represent the second frame f_2 using the first frame f_1 along with the set of source points and their strengths.

IV. Intuition behind the Radial Transform and construction of the codec

While the DCT exploits the translational invariance of the 2D signal in the transformed domain [13], the spherical construct of RT exploits the rotational invariance. Using the example pattern depicted in Fig. 3, we build the intuition behind the features that are rotationally invariant. We use the modifications explained further for the same. Note that the RT performs significantly better than the DCT as shown in Table I. The traditional video codec performs as depicted in Fig. 2. In the current exercise, the DCT blocks in the traditional codec are replaced with the spherical construct of the RT.

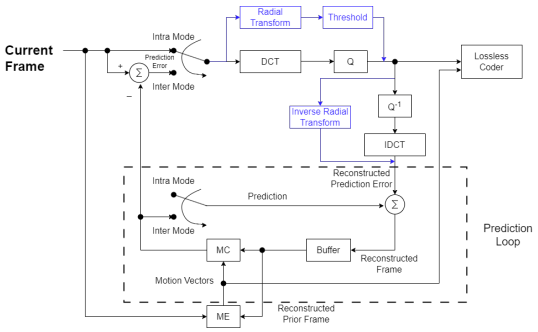


Figure 2: Video Codec Encoders

The new video codec would perform as depicted in Fig. 2 and the change is denoted in blue. The frames are split into blocks of size 8×8, leading to 64 target points and 64 source points, thus having to find 64 points on the sphere which are *distributed in a symmetric fashion*. The problem of finding such a distribution for any n points on the sphere is still an open problem, but it does have some very nice approximations. The electron simulation method [14] is used to obtain 64 points of the $\mathcal{P}$ set. The spherical structure of the transform is modified as: $\boldsymbol{\alpha} \in \mathcal{P}$ to lie on a 3 dimensional sphere with radius r. The frames during transmission are normalized. The $\Phi(\mathbf{x})$ mapping is chosen such that pixels which are close to each other in the frame are close to each other on the sphere (after the mapping). Take $\Psi_f(\mathbf{x}) = (\frac{f(\mathbf{x})}{256} + 1) \cdot \boldsymbol{\alpha} \cdot \frac{1}{r}$, where $\boldsymbol{\alpha} = \Phi(\mathbf{x})$. Figures 1b and 1c give a visual on how an 8×8 block is mapped in $3D$. In the newly constructed video codec, the transformation is applied to the residual error frame. For this purpose, the base is chosen to be the 0 matrix, that is, $f(\mathbf{x}) \equiv 0$. It follows that

$$\boldsymbol{H}_\theta = \begin{bmatrix} \|\frac{\boldsymbol{\alpha_1}}{r} - \boldsymbol{\alpha_1}\|^{-\theta} & \|\frac{\boldsymbol{\alpha_1}}{r} - \boldsymbol{\alpha_2}\|^{-\theta} & \dots & \|\frac{\boldsymbol{\alpha_1}}{r} - \boldsymbol{\alpha_k}\|^{-\theta} \\ \|\frac{\boldsymbol{\alpha_2}}{r} - \boldsymbol{\alpha_1}\|^{-\theta} & \|\frac{\boldsymbol{\alpha_2}}{r} - \boldsymbol{\alpha_2}\|^{-\theta} & \dots & \|\frac{\boldsymbol{\alpha_2}}{r} - \boldsymbol{\alpha_k}\|^{-\theta} \\ \vdots & \vdots & \ddots & \vdots \\ \|\frac{\boldsymbol{\alpha_k}}{r} - \boldsymbol{\alpha_1}\|^{-\theta} & \|\frac{\boldsymbol{\alpha_k}}{r} - \boldsymbol{\alpha_2}\|^{-\theta} & \dots & \|\frac{\boldsymbol{\alpha_k}}{r} - \boldsymbol{\alpha_k}\|^{-\theta} \end{bmatrix} \quad (4)$$

This modification gets rid of any cases when $P \cap Q \neq \emptyset$ might occur. Throughout the analysis, quantization of the transmitted block is done by dropping the digits after the second decimal point. The threshold in the new codec is parameterised by $t \in (0, 100)$. During transmission, the codec allows the values greater than $t\%$ of the maximum element of the block to pass and sets the rest of the block to 0. Note that once the matrix $\boldsymbol{H}_\theta$ is set, the inverse needs to be calculated only once. In Section V, the performance of RT is analysed while varying the hyper-parameters.

V. Results and Discussion

Analysis and comparison of the two codecs have been carried out using the standard videos in Derf's collection [15] of SD Content and Below. All the videos used in the analysis were converted to gray scale. The standard videos are encoded with different values of (r, θ) and the performance of RT and DCT is compared on three metrics: Peak Signal-to-Noise Ratio (PSNR), Multi-Scale SSIM (MSSSIM), entropy

Pattern #	MAE		Entropy		# of non-zero values	
	DCT	RT	DCT	RT	DCT	RT
1	5.14	2.14	2.34	0.337	31	4
2	5.85	2.14	2.23	0.337	25	4
3	4.78	1.98	1.89	0.337	21	4
4	6.68	2.06	2.30	0.337	24	4
5	5.65	2.07	1.87	0.337	19	4

Table I: Mean absolute error and entropy for selected patterns with $(r, \theta) = (0.8, 5)$ for RT and a quality level of 50 for DCT

of the transformed domain post quantization. Throughout the analysis, the codec sends I-frames at intervals of 15 and t is set to 8.

A. Tuning the hyper parameters

Fig. 4a shows the results for the first 300 frames of 16 standard videos selected at random from the Derf's data set. Points (r, θ) belong to a large grid with $r \in \{0.7, 0.75, 0.8, 0.85, 0.9\}$ and $\theta \in \{3, 3.5, 4, 4.5, 5, 5.5\}$.

The points $(0.8, 4)$, $(0.9, 3)$ and $(0.85, 3.5)$ are observed to give the best performance on both the PSNR and MSSSIM, the PSNR of RT being higher than its DCT counterpart (by a little under 1.5 dB) and MSSSIM inching right below the value obtained using the DCT (by around 0.002). The neighbourhoods of the points mentioned are further analysed.

B. Probing Further

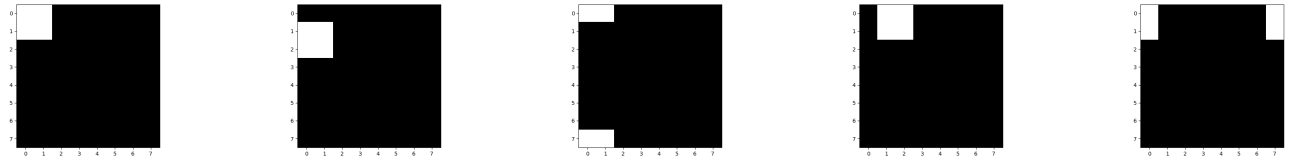
Figures 4b, 4c, 4d show the performance metrics of RT against DCT in the neighbourhood of $(0.8, 4)$, $(0.9, 3)$, and $(0.8, 5)$. For each point (r, θ), we probe 100 points with (r, θ) as the center of the grid: $r \in \{r - 0.05, r - 0.04, \dots, r + 0.04\}$ and $\theta \in \{\theta - 0.5, \theta - 0.4, \dots, \theta + 0.4\}$. The PSNR of the RT surrounding the neighbourhoods is significantly higher than that of DCT in the regime where entropy is around 0.7.

C. Neighbourhood search

All three neighbourhoods show similar PSNR and MSSSIM trends. The MSSSIM of the points in the vicinity of each of the neighbourhoods is marginally lower than the DCT (order of the third decimal). The PSNR of RT is significantly higher (36.5 dB) than DCT (35.2dB) in the region where entropy is between 0.70 and 0.85. As these points obtained from the initial grid search show similar performance, the region can be safely concluded to be a stable optimum. The performance of a few handpicked points around $(0.85, 3.5)$ run on all the standard videos is shown in Table II. A notable point is $(0.87, 3.3)$ which reflects a PSNR of 35.91 dB at entropy of 0.759 while the DCT at quantization level 70 has a PSNR of 34.38 dB at an entropy of 0.731. The MSSSIM are both of the order 0.99 and vary marginally in the third decimal.

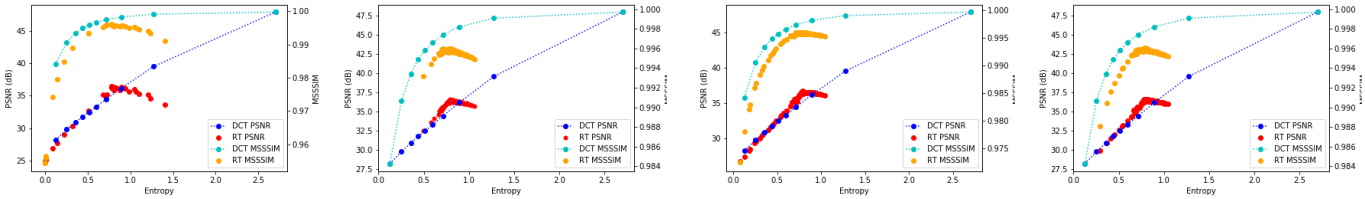
D. Visuals of reconstruction

Figures 5a, 5b and 5c show qualitative visuals of the reconstruction for each transform, along with the ground truth, from a frame drawn at random. PSNR of 30.15 dB by RT at an entropy of 2.14, and a PSNR of 28.80 dB by DCT at an entropy of 2.33, are observed. The RT has a MSSSIM of 0.987 while the DCT has a 0.993. The MAEs are 6.17 and



(a) Pattern 1 (b) Pattern 2 (c) Pattern 3 (d) Pattern 4 (e) Pattern 5

Figure 3: Example pattern to demonstrate the rotationally invariant features of the RT



(a) Initial grid search (b) Vicinity of (0.8, 4) (c) Vicinity of (0.9, 3) (d) Vicinity of (0.85, 3.5)

Figure 4: PSNR & MSSSIM vs entropy

(a) Original frame (b) RT reconstruction (c) DCT reconstruction

Figure 5: Comparison of original frame with reconstructed frames. In 5b and 5c, the left image shows the reconstructed frame and the right image shows the SSIM map. The SSIM maps indicate better reconstruction of texture rich regions by RT.

7.13 respectively. Figures 5b and 5c represent the SSIM map after reconstruction. The squared error values taken by RT are observed to be significantly lesser than the values taken by DCT. Moreover, observe that RT is able to reconstruct the texture rich regions much better than DCT.

Transform	Quality Level / Parameters	Entropy	PSNR	MSSSIM
DCT	60	0.612	33.25	0.9959
DCT	70	0.731	34.38	0.9969
DCT	80	0.809	35.10	0.9973
RT	(0.89, 3.1)	0.765	35.92	0.9943
RT	(0.87, 3.3)	0.759	35.91	0.9940
RT	(0.84, 3.7)	0.759	35.83	0.9940
RT	(0.84, 3.9)	0.730	35.50	0.9939
RT	(0.87, 3.5)	0.726	35.50	0.9940
RT	(0.89, 3.4)	0.703	34.96	0.9941

Table II: Performance metrics for the selected parameters

Remark (Escape Mechanism)**.** *At this point in time, the authors are unable to prove the existence of the inverse of the $\boldsymbol{H}_\theta$ matrix in a general setting. In all practical simulations, a singular $\boldsymbol{H}_\theta$ was never encountered. $\boldsymbol{H}_\theta$ was found to be invertible for all choices of r and θ. However, for the sake of completeness, an escape mechanism is proposed if the situation ever arises where $\boldsymbol{H}_\theta$ is non invertible: we switch to the intra mode and send the frame as a whole.*

VI. Conclusion

The results depict with clarity that the Radial Transform is significantly better than the Discrete Cosine Transform, increasing the PSNR by a massive 1.5 dB while maintaining entropy level in the regime around 0.7. From the visual qualitative analysis, it is observed that RT compresses texture much better than DCT. Going forward, a hybrid codec, which uses RT to compress the texture rich blocks and DCT to compress the smoother blocks, may lead to much better performance. Borrowing ideas from physics and employing a rotationally invariant object such as the spherical construct of the Radial Transform led to a very viable alternative transformation block in the field of video compression. Unlike DCT, RT provides a much wider range of freedom: the h function, the Φ and Ψ_f maps. In the spherical construct, (r, θ) along with t can be varied. Going forward, the Radial Transform could be an alternative in other fields of data compression, signal processing, systems engineering and much more. An exploratory endeavor to use mathematical objects which exhibit both rotational and translational invariance to build new constructs of the Radial Transform might increase the performance results shown in this paper.

References

[1] Nasir Ahmed, T. Natarajan, and Kamisetty R Rao, "Discrete cosine transform," *IEEE transactions on Computers*, vol. 100, no. 1, pp. 90–93, 1974.

[2] Zhou Wang, Eero P Simoncelli, and Alan C Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. Ieee, 2003, vol. 2, pp. 1398–1402.

[3] Brad Osgood, "The fourier transform and its applications," *Lecture notes for EE*, vol. 261, pp. 20, 2009.

[4] Stéphane Mallat, *A wavelet tour of signal processing*, Elsevier, 1999.

[5] Xun Cai and Jae S. Lim, "Algorithms for transform selection in multiple-transform video compression," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5395–5407, 2013.

[6] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on circuits and systems for video technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[7] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the h. 264/avc video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003.

[8] Didier Le Gall, "Mpeg: A video compression standard for multimedia applications," *Communications of the ACM*, vol. 34, no. 4, pp. 46–58, 1991.

[9] Vinit D. Raut and Surekha Dholay, "Analyzing image compression with efficient transforms multistage vector quantization using radial basis function neural network," in *2015 IEEE International Conference on Engineering and Technology (ICETECH)*, 2015, pp. 1–6.

[10] Zhouye Gu, Weisi Lin, Bu-Sung Lee, and Chiew Tong Lau, "Rotated orthogonal transform (rot) for motion-compensation residual coding," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4770–4781, 2012.

[11] Elena Alshina, Alexander Alshin, and Felix C. Fernandes, "Rotational transform for image and video compression," in *2011 18th IEEE International Conference on Image Processing*, 2011, pp. 3689–3692.

[12] Edward B Saff and Amo BJ Kuijlaars, "Distributing many points on a sphere," *The mathematical intelligencer*, vol. 19, pp. 5–11, 1997.

[13] Gregory K Wallace, "The jpeg still picture compression standard," *IEEE transactions on consumer electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[14] Ali Katanforoush and Mehrdad Shahshahani, "Distributing points on the sphere, i," *Experimental Mathematics*, vol. 12, no. 2, pp. 199–209, 2003.

[15] "Xiph.org Video Test Media: Derf's collection," https://media.xiph.org/video/derf/, Accessed: September 9, 2023.