DCT-Based Local Descriptor for Robust Matching and Feature Tracking in Wide Area Motion Imagery

Ke Gao, *Graduate Student Member, IEEE*, Hadi AliAkbarpour, *Member, IEEE*,
Gunasekaran Seetharaman, *Fellow, IEEE*, and Kannappan Palaniappan, *Senior Member, IEEE*

***Abstract*— We introduce a novel discrete cosine transform-based feature (DCTF) descriptor designed for both robustly matching features in aerial video and tracking features across wide-baseline oblique views in aerial wide area motion imagery (WAMI). Our DCTF descriptor preserves local structure more compactly in the frequency domain by utilizing the mathematical properties of the discrete cosine transform (DCT) and outperforms widely used the spatial-domain feature extraction methods, such as speeded up robust features (SURF) and scale-invariant feature transform (SIFT). The DCTF descriptor can be used in combination with other feature detectors, such as SURF and features from accelerated segment test (FAST), for which we provide experimental results. The performance of DCTF for image matching and feature tracking is evaluated on two city-scale aerial WAMI data sets (ABQ-215 and LA-351) and a synthetic aerial drone video data set digital imaging and remote sensing image generation (Rochester Institute of Technology (RIT)-DIRSIG). DCTF is a compact 120-D descriptor that is less than half the dimensionality of state-of-the-art deep learning-based approaches, such as SuperPoint, LF-Net, and DeepCompare, which requires no learning and is domain-independent. Despite its small size, the DCTF descriptor surprisingly produces the highest image matching accuracies ($F_1 = 0.76$ and ABQ-215), the longest maximum and average feature track lengths, and the lowest tracking error (0.3 pixel, LA-351) compared with both handcrafted and deep learning features.**

***Index Terms*— 3-D stereo, aerial video, convolutional neural network (CNN), DCT, feature descriptor, matching, point correspondences, spatial frequency.**

I. Introduction

Local keypoint feature detection, matching, and tracking are essential ingredients in many computer vision tasks. An image matching pipeline consists of three stages—feature or keypoint detection, feature descriptor representation, and descriptor matching. For feature tracking, an additional association step is needed. Good local features provide distinctive image representations that enable keypoints to be distinguishable within their neighborhoods across new views [1]. In many detect-and-track feature matching approaches, a reliable feature detector is used to accurately and consistently extract feature points, which is then used to establish correspondences or feature tracks between frames in an image sequence. A feature descriptor is an encoder that maps a feature point or region into a distinctive high-dimensional vector by incorporating the local neighborhood information. A robust feature descriptor is expected to be invariant to a range of image transformations, including translation, scale, illumination, perspective, blur, compression, and noise. The feature descriptor matching module computes distances between a reference and candidate set of feature descriptors using suitable similarity or distance metrics to establish the best match.

Good features for matching and tracking being the foundation of many computer vision algorithms have led to a proliferation of detectors [1], [2] and descriptors [3]. Scale-invariant feature transform (SIFT) [4] has established itself as one of the best-performing hand-crafted feature matching algorithms. Speeded up robust features (SURF) [5] was developed to improve upon SIFT and performs better in terms of speed and accuracy. A few commonly used feature extraction methods, such as Oriented features from accelerated segment test (FAST) and Rotated BRIEF (ORB) [6] and accelerated-KAZE (AKAZE) [7], use binary descriptors for high computational efficiency with a tradeoff in matching accuracy. Recently, new deep learning approaches for feature analysis have been developed to improve image matching pipelines [8]–[10]. However, deep learning approaches are supervised requiring a computationally expensive learning phase. Furthermore, labeled training data may be difficult to obtain and are noisy in domains, such as city-scale aerial video or self-driving vehicles under obscuration and weather.

In the aerial video, such as wide area motion imagery (WAMI) [11], accurate feature matching and tracking are the performance limiting steps for tasks, such as photogrammetry, stereo, tracking, and target recognition. Due to oblique viewing angles and perspective shape distortions, reaching subpixel accuracy to support structure from motion (SfM) and scene perception, for autonomous drone navigation [12], is challenging, especially when there are highly repetitive textures, such as roof tiles, siding, and windows in urban scenes. Most traditional and deep feature analysis operators operate in the spatial domain. We propose a novel discrete cosine transform feature (DCTF) descriptor to generate a frequency domain feature descriptor that is invariant to various image transformations. DCTF requires no training step and is significantly more compressed than recent deep learning-based approaches. The DCTF descriptor works in conjunction with

Manuscript received December 19, 2019; revised April 29, 2020; accepted May 27, 2020. This work was supported in part by the U.S. Army Research Laboratory under Grant W911NF-1820285 and in part by Army Research Office under Grant DURIP W911NF-1910181. *(Corresponding author: Ke Gao.)*

Ke Gao, Hadi AliAkbarpour, and Kannappan Palaniappan are with the CIVA Lab, Department of EECS, University of Missouri, Columbia, MO 65211 USA (e-mail: kegao@mail.missouri.edu; hd.akbarpour@gmail.com; palaniappank@missouri.edu).

Guna Seetharaman is with Advanced Computing Concepts, U.S. Naval Research Laboratory, Washington, DC 20375 USA (e-mail: guna.seetharaman@nrl.navy.mil).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2020.3000762

any detector and provides state-of-the-art feature matching and tracking performance in aerial video.

II. DCTF

In this section, we provide the basic definitions of the discrete cosine transform (DCT) as a background. We then describe the methodology used to build the novel DCTF descriptor for feature matching.

A. DCT

The DCT maps a signal from the spatial domain to the frequency domain. The DCT captures visually important spatial frequency information in a 2-D signal with only a small compact set of low-frequency coefficients that cluster in the upper left corner of the corresponding 2-D DCT matrix. Due to this energy compaction property, the DCT is universally used in data compression and image quality assessment applications.

The 2-D DCT of an $M \times N$ image matrix f is defined as

$$F(u,v) = \alpha_u \alpha_v \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i,j) \times \cos \frac{(2i+1)u\pi}{2M} \times \cos \frac{(2j+1)v\pi}{2N} \quad (1)$$

where $0 \leq u \leq (M-1)$, $0 \leq v \leq (N-1)$, $f(i,j)$ is the pixel intensity, and $F(u,v)$ is the transform coefficient at row u and column v in the DCT matrix. Scalars α_u and α_v are the normalization coefficients defined as

$$\alpha_u = \begin{cases} 1/\sqrt{M}, & u = 0 \\ \sqrt{2/M}, & 1 \leq u \leq M-1 \end{cases} \quad (2)$$

$$\alpha_v = \begin{cases} 1/\sqrt{N}, & v = 0 \\ \sqrt{2/N}, & 1 \leq v \leq N-1. \end{cases} \quad (3)$$

The coefficient $F(0,0)$ at the top left corner in the DCT matrix is the dc term (coefficient). The rest of the DCT coefficients are ac terms that correspond to high spatial frequency coefficients in increasing order. Using the natural properties of the DCT, the proposed DCTF descriptor can achieve invariance to photometric transformations. According to (1), the dc term $F(0,0)$ of the DCT matrix for a 2-D image f can be computed as

$$F(0,0) = \alpha_u \alpha_v \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i,j) \quad (4)$$

which can be interpreted as the sum of all pixel (graylevel) intensities in image f. Illumination differences can be normalized by dividing each ac term by the dc term $(F(u,v)/F(0,0))$. Other photometric transformations, such as image blur and JPEG compression artifact, can be handled by using the low-frequency coefficients in a DCT matrix and disregarding the high-frequency ones.

B. Feature Descriptor Representation and Matching

For each feature keypoint $\mathbf{p}_k$ detected in the input image, s square image patches of different sizes are center-cropped around $\mathbf{p}_k$. Let $M_0 \times M_0$ be the dimension of the smallest crop patch. For each $i \in \{0, 1, 2, \ldots, s-1\}$, we crop an image patch $\mathbf{p}_k^{(i)}$ of size $\alpha^i M_0 \times \alpha^i M_0$, where α is the scaling factor between patch sizes. The DCT of each center-cropped patch, $\mathbf{C}_i = \mathrm{DCT}(\mathbf{p}_k^{(i)})$, is used to encode the most informative spatial frequency information. Since most of the visually significant information with photometric invariance is stored at the upper left corner of $\mathbf{C}_i$, we use the zig-zag scan shown in Fig. 1 to reorganize the DCT coefficients and keep the dc term and the first n ac terms. We divide the n selected ac terms by the dc term to normalize for illumination changes. We repeat the process for each nested crop patch $\mathbf{p}_k^{(i)}$ and concatenate the selected coefficients into a 1-D vector $\boldsymbol{v}_k$ to form the DCTF feature descriptor for keypoint $\mathbf{p}_k$. The concatenation operation is defined as

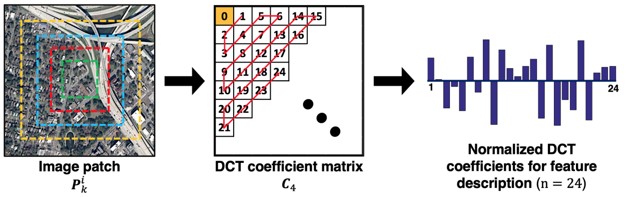

Fig. 1. (Left) Five center-cropped (nested) image patches around a keypoint $\mathbf{p}_k$ are shown as colored dashed boxes. (Middle) DCT coefficient matrix $\mathbf{C}_4$ for the largest crop patch, $\mathbf{p}_k^{(4)}$. The dc term is highlighted in yellow color, and the rest are the selected ac terms. (Right) Zig-zag scan is applied, and the first 24 ac terms are selected and normalized ($\mathbb{Z}_{24}$ operator). Concatenating from all five patches forms the DCTF feature descriptor representation, $\boldsymbol{v}_k$.

$$\boldsymbol{v}_k = \bigcup_{i=0}^{s-1} \mathbb{Z}_n[\mathbf{C}_i] = \bigcup_{i=0}^{s-1} \mathbb{Z}_n\left[\mathrm{DCT}(\mathbf{p}_k^{(i)})\right] \quad (5)$$

where the zig-zag operator $\mathbb{Z}_n[\mathbf{C}_i]$ extracts the first $(n+1)$ elements of $\mathbf{C}_i$ in zig-zag scan order and then normalizes the transform coefficients (dividing by $\mathbb{Z}_0$ dc term) and keeping n ac terms. Using patches of different sizes around a keypoint allows us to encode the distinctive characteristics from various immediate neighborhoods, which improves the robustness of matching. In the experiments, we use five crops ($s = 5$) starting at a patch size with $M_0 = 16$, scaling factor $\alpha = 1.5$, and the largest patch size of 81×81. The DCTF descriptor, $\boldsymbol{v}_k$, is a 120-D vector using $n = 24$ coefficients and $s = 5$ crops.

```
Algorithm 1 Generate the DCTF Descriptor
Require: Grayscale image $I$, feature keypoint $\mathbf{p}_k$.
Ensure: DCTF descriptor $\boldsymbol{v}_k$.
  Initialize $\boldsymbol{v}_k$ and scaling factor $\alpha$.
  for $i \leftarrow 0$ to $(s-1)$ do
    Crop a $\alpha^i M_0 \times \alpha^i M_0$ patch $\mathbf{p}_k^{(i)}$ from $I$ centered at $\mathbf{p}_k$.
    Compute DCT coefficient matrix $\mathbf{C}_i$ for $\mathbf{p}_k^{(i)}$.
    // Zig-zag scan $\mathbf{C}_i$
    $\mathbb{Z}_n[\mathbf{C}_i] \leftarrow$ Normalize the first $n$ ac terms.
    $\boldsymbol{v}_k \leftarrow \boldsymbol{v}_k \oplus \mathbb{Z}_n[\mathbf{C}_i]$. // Concatenation
  end for
```

Taking advantage of the mathematical properties of the DCT leads to a local frequency domain feature descriptor with invariance to photometric transformations in the image. The proposed DCTF descriptor requires no training process and imposes no constraints on the type or size of input data, making it more compatible than its deep learning-based counterparts. To establish correspondences between feature keypoints in the reference image and those in the matching

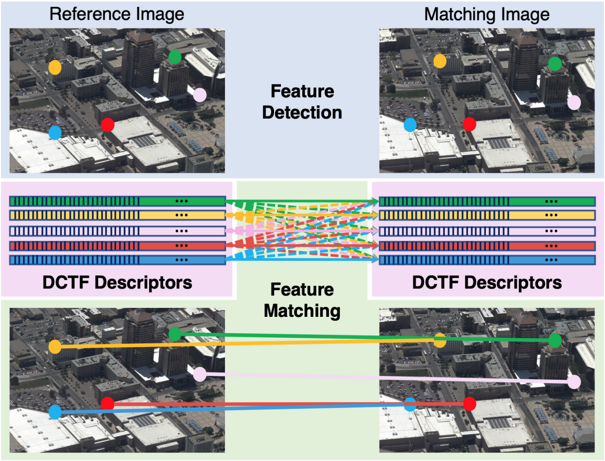

Fig. 2. Feature matching pipeline. DCTF descriptor is generated for each keypoint in both reference and matching images. Each reference descriptor is matched against all the matching candidate descriptors, and the best match is determined using the DR matching scheme. Keypoint correspondences between two images are shown as color-coded solid lines.

image, we use the distance ratio (DR) matching strategy [4]. For each descriptor in the reference image, we determine its nearest neighbor (NN) descriptor $n1$ from all candidate keypoints in the matching image based on L_2 feature distance. The nearest d_{n1} and second NN d_{n2} distances are used to check the ratio, $\rho = d_{n1}/d_{n2}$. The NN $n1$ is considered as the best match for the reference descriptor if the ratio ρ is below a threshold ($\rho < 0.7$ in this letter). The feature matching pipeline is illustrated in Fig. 2.

III. Experimental Results

We evaluate the proposed DCTF feature descriptor on two city-scale aerial WAMI data sets from TransparentSky and one synthetic aerial drone video sequence from Rochester Institute of Technology (RIT) [13]. The DCTF descriptor used in combination with SURF [5] feature detector is denoted as SURF+DCTF and with the FAST feature detector [14] as FAST+DCTF. Note that DCTF is only a feature descriptor and requires a separate feature detector. We compare DCTF to four traditional approaches—SURF, SIFT, AKAZE, and ORB—and three deep learning approaches—DeepCompare, LF-Net, and SuperPoint. All of the traditional feature extraction methods, including DCTF, were implemented in C++ using OpenCV 2.4 libraries. The pretrained models for the deep learning methods are provided by Zagoruyko and Komodakis [8], DeTone *et al.* [9], and Ono *et al.* [10].

A. Image Matching Accuracy in Aerial WAMI (ABQ-215)

We evaluate the feature matching performance of the proposed method on the first ten frames from the Albuquerque (ABQ-215) WAMI sequences [11], [15]. The high-resolution aerial images (6600 × 4400 pixels) were captured using an airborne platform with a gimbal-mounted camera system that tracked the center of the scene. The first and tenth frames are shown in Fig. 3, with a viewing angle difference of about 1.67° between each adjacent pair of images. The full flight trajectory (215 frames) covers a complete orbit with a radius of 2.5 km. The typical range between the camera location and the scene center was 3 km. The height above ground level (AGL) was 1.5 km, and the average ground sampling distance (GSD) was 20 cm. The bundle adjusted metadata (camera poses) for the image sequence is used as the ground-truth fundamental matrix for evaluating the accuracy of feature correspondence tracking [16], [17].

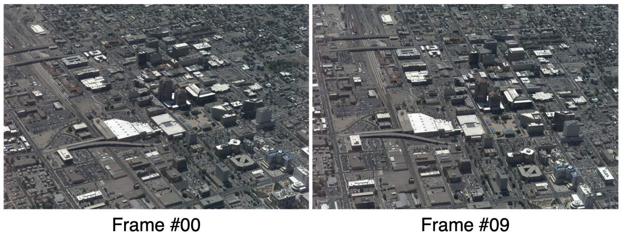

Fig. 3. Two sample frames from the ABQ-215 WAMI data set. Image size is 6600 × 4400.

We first perform feature keypoint detection on the reference image that is the first frame in the sequence (frame #00 in Fig. 3) and then generate feature descriptor for each keypoint. The same procedure is applied to the following nine consecutive frames with increasing amount of perspective transformations. After that, we adopt the DR matching strategy to match keypoints in the reference image against those in each matching image. Correct matches are identified using the ground truth. The matching results are evaluated in terms of the F-measure (F_1)

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{6}$$

where recall and precision are defined as follows:

$$\text{recall} = \frac{\#\text{correct matches}}{\#\text{correspondences}} \tag{7}$$

$$\text{precision} = \frac{\#\text{correct matches}}{\#\text{correct matches} + \#\text{false matches}}. \tag{8}$$

The evaluation results are presented in Table I. The upper half of Table I demonstrates the performance of DCTF using different numbers of DCT coefficients n from each image crop around a keypoint for generating a descriptor (discussed in Section II-B). We use a fixed number of five crops ($s = 5$) and evaluate three different DCT coefficients $n = \{12, 24, 36\}$ for DCTF descriptors of three sizes ($5 \times 12 = 60$, $5 \times 24 = 120$, and $5 \times 36 = 180$). Combined with either SURF keypoints or FAST keypoints, the 120-D DCTF descriptor provided the best results compared with 60-D and 180-D. It is important to determine the optimal size of the DCTF descriptor, especially for large-scale aerial imagery, because incorporating too few DCT coefficients will make the descriptor less discriminative, while incorporating too many will introduce redundant high-frequency information. We use the 120-D DCTF descriptor as the default in this letter.

Overall, the DCTF descriptor exhibits higher matching accuracy than the other approaches, including state-of-the-art deep learning feature methods. SURF+DCTF outperforms the rest by a large margin. FAST+DCTF produces the second-best matching performance. Note that LF-Net and SuperPoint could not handle the large size of aerial imagery, and both methods did not provide any results on the ABQ-215 WAMI data set. Surprisingly, the widely used ORB feature

TABLE I

FEATURE MATCHING EVALUATION USING F_1 FOR THE AERIAL ABQ-215 WAMI DATA SET USING 2000 FEATURES PER FRAME BASED ON DR MATCHING (RATIO 0.7). T1–T9 CORRESPOND TO OBLIQUE MULTIVIEW FRAMES #01 TO #09, WITH MATCHING TO REFERENCE FRAME #00, BASED ON BUNDLE ADJUSTED CAMERA POSE AS THE GROUND TRUTH. BEST RESULTS ARE HIGHLIGHTED IN BOLD AND SECOND BEST IN ITALICS. LF-NET AND SUPERPOINT FAILED ON THIS DATA SET

Method	T1	T2	T3	T4	T5	T6	T7	T8	T9
SURF+DCTF (60-D)	**0.76**	**0.75**	*0.62*	0.52	*0.38*	*0.28*	*0.20*	**0.13**	*0.09*
SURF+DCTF (120-D)	**0.76**	**0.75**	**0.63**	**0.55**	**0.42**	**0.29**	**0.21**	0.11	**0.10**
SURF+DCTF (180-D)	**0.76**	**0.75**	*0.62*	*0.53*	*0.38*	0.25	0.17	0.08	0.06
FAST+DCTF (60-D)	*0.63*	0.57	0.50	0.38	0.25	0.16	0.09	0.05	0.03
FAST+DCTF (120-D)	*0.63*	*0.59*	0.51	0.41	0.28	0.16	0.09	0.03	0.02
FAST+DCTF (180-D)	*0.63*	0.58	0.50	0.39	0.23	0.12	0.06	0.03	0.01
SURF	0.62	0.58	0.48	0.37	0.27	0.21	0.15	*0.12*	0.08
SIFT	0.52	0.48	0.40	0.34	0.27	0.17	0.15	0.09	0.07
AKAZE	0.47	0.40	0.29	0.22	0.14	0.09	0.06	0.03	0.03
ORB	0.10	0.07	0.05	0.03	0.01	0.01	0.01	0.01	0.01
DeepCompare	0.59	0.54	0.45	0.34	0.25	0.17	0.11	0.07	0.05

TABLE II

EVALUATION OF FEATURE TRACKING ON THE LA-351 DATA SET USING 5000 FEATURES PER FRAME. EEE USING THE BUNDLE ADJUSTED CAMERA POSE AS GROUND TRUTH (10) IS CALCULATED IN PIXELS. MATCHING STRATEGY USED IS DR OR NN OR THRESHOLD. BEST RESULTS ARE HIGHLIGHTED IN BOLD AND SECOND BEST IN ITALICS

Method	Matching Strategy	# of Tracks	Avg Track Length	Max Track Length	EEE Mean	EEE Std
SURF+DCTF	DR (0.7)	161,448	**9.13**	**350**	**0.30**	*1.53*
FAST+DCTF	DR (0.7)	190,697	7.45	225	0.39	2.79
SURF	DR (0.7)	235,425	5.96	148	0.45	6.67
SIFT	DR (0.8)	242,696	6.32	164	0.76	10.82
AKAZE	DR (0.8)	160,670	*8.88*	*349*	0.70	10.35
ORB	NN	239,395	6.56	160	16.21	52.55
DeepCompare	Threshold	129,774	2.59	33	0.75	11.58
LF-Net	NN	195,776	8.02	316	3.52	25.71
SuperPoint	NN	187,228	7.45	308	*0.38*	**0.63**

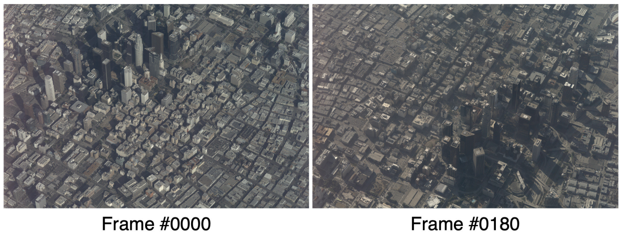

Fig. 4. Sample frames from the LA-351 WAMI data set. The aerial video consists of 351 images, each 1650 × 1100 pixels.

for real-time simultaneous localization and mapping (SLAM) applications does not perform well on aerial imagery. The FAST detector performs well for small viewpoint changes, but performance quickly degrades as the perspective distortions increased (beyond T4).

B. Feature Tracking Accuracy in Aerial WAMI (LA-351)

To evaluate the performance of the DCTF descriptor in the context of feature tracking on long aerial video data set, we use the Los Angeles (LA-351) WAMI sequence [11] that consists of 351 high-resolution images (see Fig. 4). Data acquisition was similar to ABQ-215 WAMI with a flight trajectory orbit radius of 4.3 km and a height AGL of 4.5 km. Note that the image size was subsampled to 1650 × 1100 pixels so that LF-Net and SuperPoint can work on this sequence even though DCTF does not have such a resolution limitation.

Accurate and robust feature tracking is critical for computer vision tasks, such as bundle adjustment (BA) and SfM [18]. Having longer feature tracks along a sequence can significantly contribute to the robustness of an SfM or BA algorithm [16]. This is because a longer group of feature matches can tie more views together and provide more constraints in the least-squares optimization methods. Feature tracks can be generated by extracting features (keypoint detection and description) in each frame, applying pairwise keypoint matching between adjacent frames, and associating the matched keypoints across the sequence. Each track τ_j is defined as the set of cameras or views

$$\tau_j = \{h_j, h_j + 1, h_j + 2, \ldots, h_j + \gamma_j - 1\} \qquad (9)$$

where γ_j is the number of consecutively matched features along with the track sequence starting from view (camera) h_j and terminating at view $h_j + \gamma_j - 1$. We compute the Euclidean epipolar error (EEE) using the bundle adjusted cameras as the ground-truth transformations to evaluate the accuracy of feature tracks [16]. To obtain the best results from each method, we use their respective default matching strategies [4]–[10]. The EEE of a feature track, τ_j, is defined as

$$\text{EEE}(j) = \frac{1}{(\gamma_j - 1)} \sum_{i=h_j}^{h_j+\gamma_j-1} d(\mathbf{p}_{i+1,j}\ , \mathbf{F}_{(i,i+1)}\mathbf{p}_{i,j}) \qquad (10)$$

where $\{\mathbf{p}_{i,j}, \mathbf{p}_{i+1,j}\}$ defines a pair of matched features between two adjacent cameras i and $i + 1$ in track τ_j, $\mathbf{F}_{(i,i+1)}$ is the ground-truth fundamental matrix that maps 2-D image points from view i to view $i + 1$, and $d(\cdot, \cdot)$ computes the perpendicular distance between the matched point and the epipolar line. Table II shows the mean and standard deviation of EEE over all feature tracks for each method, as well as total number of tracks, average track length, and maximum track length. DCTF using SURF detected feature keypoints generated the longest feature tracks, the smallest average EEE track error, and the second smallest track error variance. DCTF using FAST keypoints also outperformed all of the traditional and deep learning methods tested, except SuperPoint that showed the smallest track localization pixel error variance.

C. Feature Tracking Accuracy in RIT DIRSIG Data Set

RIT digital imaging and remote sensing image generation (DIRSIG)-simulated drone video data set [13] consists of synthetic aerial imagery and camera metadata (see Fig. 5). The simulated flight trajectory is a full orbit around the scene, and the nominal declination angle was 40°. The synthetic data set consists of 420 RGB synthetic aerial images. The image size is 1200×800 with 32-μm square pixel size and a focal length of 125 mm [13]. Similar to the LA-351 WAMI data set, we tested feature tracking on the RIT DIRSIG data set and used

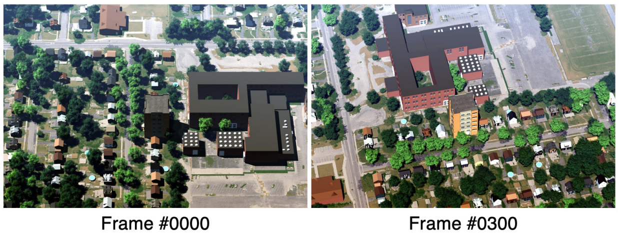

Fig. 5. Sample frames from the RIT DIRSIG simulated drone data set. The aerial video consists of 420 RGB synthetic images, each 1200 × 800 pixels.

TABLE III

EVALUATION OF FEATURE TRACKING ON THE RIT DIRSIG DATA SET USING 3000 FEATURES PER FRAME. SPEED OF EACH METHOD IS SHOWN IN TERMS OF TIME TO PROCESS ONE FRAME ON AN INTEL CORE i7-7700HQ CPU. BEST RESULTS ARE HIGHLIGHTED IN BOLD AND SECOND BEST IN ITALICS

Method	Descriptor Size	Matching Strategy	Max Track Length	EEE Mean	EEE Std	Time per Frame (s)
SURF+DCTF	120 (floats)	DR (0.7)	**420**	**0.23**	*0.99*	0.96
FAST+DCTF	120 (floats)	DR (0.7)	260	*0.41*	**0.81**	0.77
SURF	64 (floats)	DR (0.7)	257	0.62	6.62	0.50
SIFT	128 (floats)	DR (0.8)	298	1.52	12.49	0.99
AKAZE	61 (bytes)	DR (0.8)	387	0.43	5.60	**0.16**
ORB	32 (bytes)	NN	206	8.96	27.30	*0.20*
DeepCompare	256 (floats)	Threshold	47	9.68	37.72	5.38
LF-Net	256 (floats)	NN	246	0.75	5.07	0.51
SuperPoint	256 (floats)	NN	*412*	0.48	1.43	1.12

EEE errors for evaluation. Table III includes feature descriptor size for each method, EEE mean and standard deviation, and time to process each frame. DCTF combined with SURF keypoints provides the best feature tracking results with the lowest EEE mean and standard deviation and the longest feature track whose length is the full orbit of the data set. SURF+DCTF has significantly less pixel error that is more than two times better than SuperPoint, which is a state-of-the-art deep learning method. FAST+DCTF produces the second-best performance in terms of accuracy. The DCTF descriptor size is more than two times smaller compared with deep learning approaches, making it suitable for many embedded applications that have limited memory or limited network bandwidth. AKAZE and ORB use a binary descriptor that is the smallest and fastest to compute but is much less accurate than DCTF. The speed of DCTF is comparable to SuperPoint and SIFT.

IV. CONCLUSION

DCTF is a novel frequency-domain descriptor using local nested image patches for accurately matching features. The 120-D DCTF with 81 × 81 patch size worked best for large-scale aerial imagery with roughly constant or slowly changing altitudes. The DCTF descriptor size is not only significantly smaller than recent deep learning algorithms but surprisingly outperforms them, including the SuperPoint deep neural network architecture, which was designed specifically to optimize subpixel matching accuracy. DCTF also outperformed the ORB feature tracker that is widely used for SLAM applications. DCTF was the best performing method for image matching on the aerial ABQ-215 WAMI data set. The DCTF descriptor provided consistent performance using SURF detectors. Similar to the other methods, DCTF also exhibited a sharp falloff in performance as image perspective distortions increased, especially using FAST features. For feature tracking, DCTF had the lowest error, longest maximum, and average track lengths on both the RIT simulated drone and LA-351 WAMI aerial data sets. Future work is focused on extending the DCTF descriptor to be invariant to rotation and scale.

ACKNOWLEDGMENT

Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the U.S. Government or Agency thereof. Wide area motion imagery (WAMI) aerial data sets were provided by Steve Suddarth at TransparentSky.

REFERENCES

[1] T. Tuytelaars and K. Mikolajczyk, “Local invariant feature detectors: A survey,” *Found. Trends Comput. Graph. Vis.*, vol. 3, no. 3, pp. 177–280, 2007.

[2] F. Bunyak, K. Palaniappan, S. K. Nath, and G. Seetharaman, “Flux tensor constrained geodesic active contours with sensor fusion for persistent object tracking,” *J. Multimedia*, vol. 2, no. 4, p. 20, Aug. 2007.

[3] C. Leng, H. Zhang, B. Li, G. Cai, Z. Pei, and L. He, “Local feature descriptor for image matching: A survey,” *IEEE Access*, vol. 7, pp. 6424–6434, 2019.

[4] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[5] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF: Speeded up robust features,” in *Proc. Eur. Conf. Comput. Vis.* Graz, Austria: Springer, 2006, pp. 404–417.

[6] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[7] P. Alcantarilla, J. Nuevo, and A. Bartoli, “Fast explicit diffusion for accelerated features in nonlinear scale spaces,” in *Proc. Brit. Mach. Vis. Conf.*, 2013, p. 1.

[8] S. Zagoruyko and N. Komodakis, “Learning to compare image patches via convolutional neural networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4353–4361.

[9] D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperPoint: Self-supervised interest point detection and description,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 224–236.

[10] Y. Ono, E. Trulls, P. Fua, and K. M. Yi, “LF-Net: Learning local features from images,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 6234–6244.

[11] K. Palaniappan, R. M. Rao, and G. Seetharaman, “Wide-area persistent airborne video: Architecture and challenges,” in *Distributed Video Sensor Networks*. London, U.K.: Springer-Verlag, 2011, pp. 349–371.

[12] H. Aliakbarpour, K. Palaniappan, and G. Seetharaman, “Stabilization of airborne video using sensor exterior orientation with analytical homography modeling,” in *Machine Vision and Navigation*. New York, NY, USA: Springer, 2020, pp. 579–595.

[13] D. Nilosek, D. J. Walvoord, and C. Salvaggio, “Assessing geoaccuracy of structure from motion point clouds from long-range image collections,” *Opt. Eng.*, vol. 53, no. 11, Nov. 2014, Art. no. 113112.

[14] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in *Proc. Eur. Conf. Comput. Vis.* Graz, Austria: Springer, 2006, pp. 430–443.

[15] R. Porter, A. Fraser, and D. Hush, “Wide-area motion imagery,” *IEEE Signal Process. Mag.*, vol. 27, no. 5, pp. 56–65, Sep. 2010.

[16] H. AliAkbarpour, K. Palaniappan, and G. Seetharaman, “Parallax-tolerant aerial image georegistration and efficient camera pose refinement—Without piecewise homographies,” *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4618–4637, Aug. 2017.

[17] H. Aliakbarpour, K. Palaniappan, and G. Seetharaman, “Robust camera pose refinement and rapid SfM for multiview aerial imagery—Without RANSAC,” *IEEE Trans. Geosci. Remote Sens.*, vol. 12, no. 11, pp. 2203–2207, Nov. 2015.

[18] J. L. Schonberger and J.-M. Frahm, “Structure-from-Motion revisited,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4104–4113.