NAVIGATION

A seasonally invariant deep transform for visual terrain-relative navigation

Anthony T. Fragoso[1]*, Connor T. Lee[1], Austin S. McCoy[1], Soon-Jo Chung[1,2]

Visual terrain-relative navigation (VTRN) is a localization method based on registering a source image taken from a robotic vehicle against a georeferenced target image. With high-resolution imagery databases of Earth and other planets now available, VTRN offers accurate, drift-free navigation for air and space robots even in the absence of external positioning signals. Despite its potential for high accuracy, however, VTRN remains extremely fragile to common and predictable seasonal effects, such as lighting, vegetation changes, and snow cover. Engineered registration algorithms are mature and have provable geometric advantages but cannot accommodate the content changes caused by seasonal effects and have poor matching skill. Approaches based on deep learning can accommodate image content changes but produce opaque position estimates that either lack an interpretable uncertainty or require tedious human annotation. In this work, we address these issues with targeted use of deep learning within an image transform architecture, which converts seasonal imagery to a stable, invariant domain that can be used by conventional algorithms without modification. Our transform preserves the geometric structure and uncertainty estimates of legacy approaches and demonstrates superior performance under extreme seasonal changes while also being easy to train and highly generalizable. We show that classical registration methods perform exceptionally well for robotic visual navigation when stabilized with the proposed architecture and are able to consistently anticipate reliable imagery. Gross mismatches were nearly eliminated in challenging and realistic visual navigation tasks that also included topographic and perspective effects.

INTRODUCTION

Remotely sensed database imagery is a common ground-truth map for visual terrain-relative navigation (VTRN). Onboard cameras are passive sensors ideal for size-, weight-, and power-constrained platforms, and extensive coverage of high-resolution imagery makes VTRN essential in the absence of Global Navigation Satellite System (GNSS) capability. Database imagery has been used to provide absolute position measurements in extraterrestrial robotic entry, descent, and landing missions (*1*, *2*); GNSS-denied defense applications (*3*); backup uncrewed aerial vehicle (UAV) state estimation (*4*); and offline subpixel geolocation of remotely sensed data products that can be extremely sensitive to localization errors (*5*).

VTRN and geolocation against target images are applications of the more general image registration problem(*6*) in which images taken from different poses, at different times, or with different sensors are transformed into the same coordinate system. Under ideal conditions, image registration is well studied and has a number of mature automatic solutions. Examples include intensity-based template matching with normalized cross-correlation (NCC) (*7*), mutual information (MI) similarity metrics (*8*), frequency-domain techniques (*9*), and feature matching (*10*). Classical registration algorithms are also often equipped with geometric and radiometric invariances that greatly simplify the VTRN problem itself. For example, feature-based methods can accommodate nonrigid image transformations due to terrain, with scale-invariant feature transform (SIFT) descriptors (*10*), particularly, being invariant to scale, two-dimensional rotation, and linear illumination changes.

In principle, an aerospace robot can be localized to within a few centimeters relative to a database using onboard imagery and a subpixel-accurate registration algorithm. High-quality georeferenced terrestrial imagery is updated on a regular basis and often available at resolutions of 10 cm per pixel or better—global coverage is available commercially at approximately 30 cm per pixel (*11*). In practice, however, aerospace robots that rely on vision regularly encounter severe appearance changes, such as snow cover or leaf drop, that change the texture, illumination, and content of the landscape beneath them. These changes violate the heuristic radiometric assumptions of classical registration and lead to fragility. Manual selection and matching of structures and control points with a stable appearance remain a respected, although time-consuming, practice for offline registration (*12*). Autonomous platforms, however, must reliably perform matching without human intervention and have historically relied on comparing radar altimetry (*13*) to a topographic database. Although topographic data are more stable than visible data, terrain matching is less accurate than image registration and exhibits poor performance at low altitudes or in flat areas (*14*).

To address the shortcomings of classical approaches, a natural option is to consider deep-learning approaches that fit stable, high-level features (*15*). Although seasonal changes in aerial imagery have received minimal attention, some deep learning techniques have been successful for challenging fusion and registration tasks, particularly in medical imaging (*16*). A common approach is to train a deep similarity metric to replace fragile classical metrics using a twinned (formerly "Siamese") architecture (*17*). For remote sensing, (*18*) learned a similarity score between synthetic aperture radar and optical images using a pseudo-twinned architecture, with separate fully convolutional networks (FCNs) for each, fed into a common comparison network. Registration in this manner requires small rigid image "chip" patches to be repeatedly sampled from larger images and passed through the network consecutively, which precludes real-time use.

[1]Division of Engineering and Applied Science, California Institute of Technology, 1200 E California Blvd., Pasadena, CA 91125, USA. [2]Jet Propulsion Laboratory (JPL), California Institute of Technology, 4800 Oak Grove Dr., Pasadena, CA 91109, USA.
*Corresponding author. Email: afragoso@caltech.edu

End-to-end networks accommodate nonrigid registration and avoid repeated evaluation of image chips by directly estimating the geometric transformation between two input images but lack the engineered advantages of classical approaches. Pure end-to-end approaches require explicit exposure to all expected transformations in training for which examples of real nonrigid transformations are exceptionally difficult to obtain. Reliability and uncertainty for end-to-end networks are also extremely difficult to interpret. Furthermore, deep learning does not necessarily outperform hand-engineered approaches for monomodal registration and is often complementary. (*19*) observed that monomodal registration of lung imagery under large deformations benefited from the hybrid use of hand-engineered and learning-based descriptors. Similarly, (*20*) augmented SIFT features with robust "deep features" taken from the intermediate activations of a pretrained VGG-16 network. Semantic features can also be extracted and matched using pixel-level segmentation, in which stable predefined structures such as road networks (*21*), lunar crater rims (*2*), or prescribed semantic elements (*22*) are labeled using a trained network and matched to a reference image. Segmentation techniques identify stable structures but require extraordinarily tedious human annotation. Structure classes must also be unambiguous and consistently distributed in imagery to be useful and are prescribed rather than themselves learned.

The registration robustness problem can also be posed using domain adaptation theory, in which a nonannotated target data "domain" supplements an annotated source domain. The source and target domains are assumed to share content relevant to a task but differ in their domain-specific statistics—for example, leaf-on and leaf-off imagery are both useful for navigation, although their vegetation patterns are different. Relevant content can be identified automatically by mapping two or more source domains to a common "latent" domain that is optimized to complete a task but in which the original source domains can no longer be distinguished. The goal is usually to assimilate unlabeled data into a model as efficiently as possible, which has seen widespread use in remote sensing primarily to improve classifier robustness while limiting annotation load (*23*). Automatic image registration, however, has received little attention in the domain adaptation literature despite its extreme sensitivity. The domain adaptation work most relevant to VTRN is based on image-to-image translation and comes from the automotive community, including adaptation for scene segmentation (*24*) and translation of degraded operating conditions into ordinary conditions (*25*).

Main contribution

In light of these observations, we derive an image transform approach to VTRN that combines the success of deep learning for image translation and domain adaptation with the well-known engineered properties of classical image registration. Rather than attempt to generate an opaque positioning estimate using deep learning and extensive annotation, we rely on the fact that conventional registration techniques in principle have perfect performance when their radiometric input assumptions are exactly met. Accordingly, we use deep learning only to modify the appearance of input images, which is a narrowly defined task at which it excels. The basis of our technique is an FCN that serves as a preprocessing step and transforms input images to a seasonally invariant domain. This network identifies and enhances stable structures, serves as an attention mechanism, and is optimized for robust performance over any well-posed classical registration algorithm. After sufficient training, a single transform allows leaf-on, leaf-off, and snow-cover test images to have an identical appearance and registration response and can mitigate some higher-frequency appearance changes, such as deep shifting shadows, that were not explicitly anticipated or trained over in advance. The transform structure lends seasonal invariance to existing conventional code and techniques without further modification or manual annotation. The result is a VTRN pipeline that inherits geometric invariances from conventional registration without explicit exposure in training while also relaxing the widely violated input assumptions that cause them to break.

RESULTS

In this section, we demonstrate the effectiveness of our deep transform architecture for optimizing area-based and feature-based image registration in challenging seasonal conditions. After describing our architecture and the datasets used for training and testing, we provide experimental performance evaluation results.

Navigation architecture

Our transform serves as an upstream preprocessing step that adds seasonal robustness to downstream VTRN or registration algorithms, which are themselves unmodified. The network is trained in advance using diverse seasonal imagery examples and deployed with locked weights, either aboard an aircraft or space robot (VTRN; Fig. 1) or for general image registration. Input and output image sizes are equal, but outputs are grayscale as is typically required for registration. We train our transform using publicly available data that are already coregistered and require no further annotation. Suitable training imagery is widely available with global and extraterrestrial coverage at a high resolution and captures years of regular appearance changes.

During training, we expose a U-Net (*26*) image transform model to matching cross-seasonal image pairs in twinned fashion—a single transform is identically shared between two parallel streams, with registration performance between the outputs used as a loss function to optimize the transform weights (Fig. 2).

At runtime, a single stream of incoming imagery, such as a navigation camera (NAVCAM) image or an unregistered scientific image product, is intercepted and replaced with transform output. This output is passed, along with a previously transformed reference image, to the rest of the registration pipeline to calculate the geometric transformation between the two images. For change detection and other scientific applications in which image appearance must be preserved, the original input may simply be warped by the now-known geometric transformation.

VTRN backend and reference image selection

The deep transform architecture is agnostic to the registration backend as long as the matching score is compatible with the one used in training. For patch matching tasks, we use a sliding window backend because of its simplicity and maturity as a VTRN technique. More efficient implementations can be accommodated with no changes to the transform, and in general, the matching backend should be selected to maximize performance.

For any VTRN architecture, reference images must also be proposed before the registration step. For clarity, we assume that black-box visual-inertial odometry is available to an accuracy sufficient only for the selection of a large reference image. To exhibit the

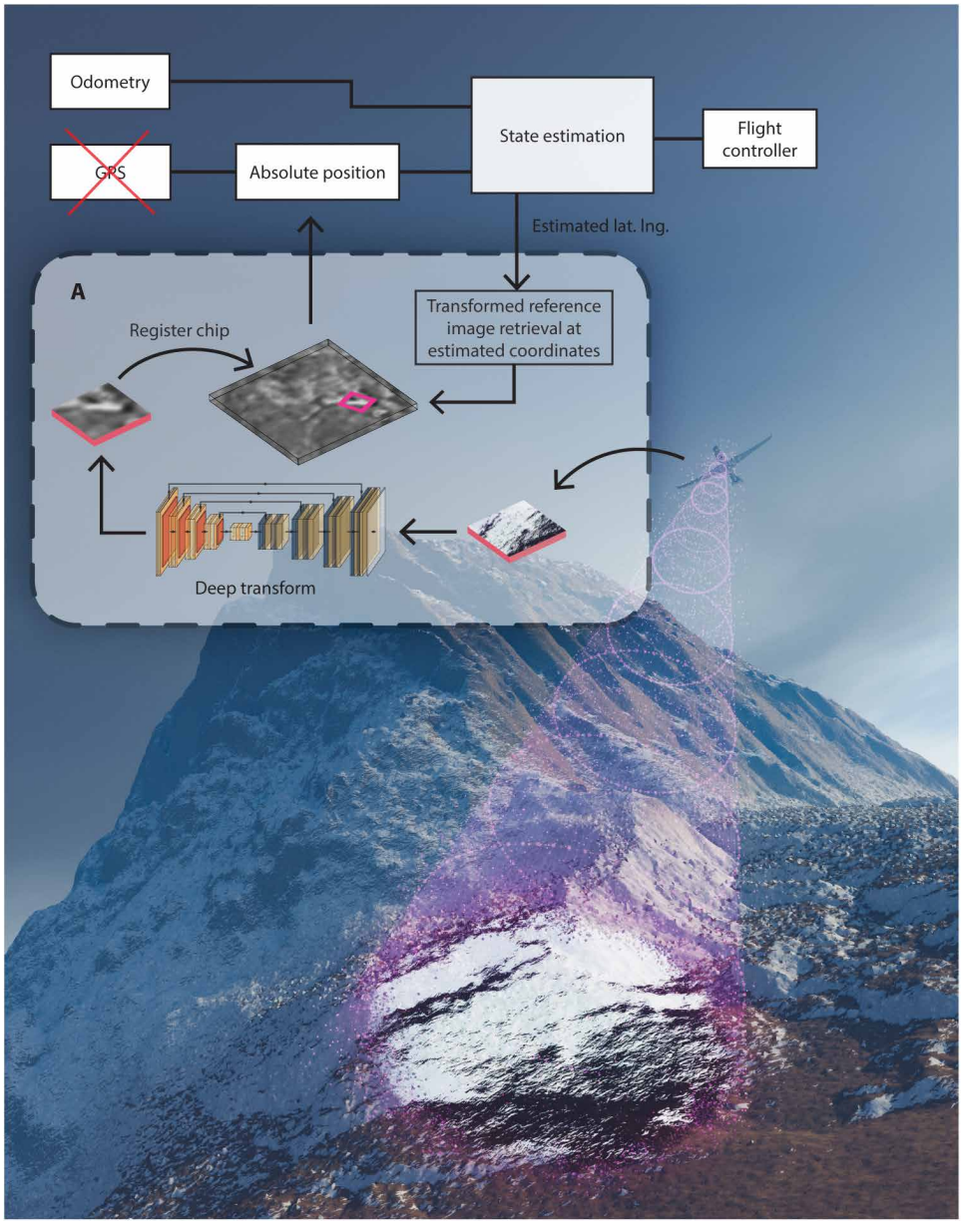

Fig. 1. VTRN using the seasonally invariant deep transform in a GNSS-denied environment. The UAV determines its absolute position by registering an online deep-transformed image of the ground into a previously deep-transformed georeferenced database image proposed using a running black-box odometry estimate. The deep transform module (**A**) removes ephemeral character from both images and forces them to satisfy the radiometric assumptions of a hand-engineered registration module that follows. As a result, the UAV can reliably recover its position from the geometric transformation between the two images using legacy techniques that fail otherwise. Network diagram generated using (*36*).

advantages of a deep transform, we also assume that odometry cannot locate the onboard image within the selected reference and consider matching independently. At the scales (images are about 1 km on a side) and update frame rates (about 20 Hz and greater) considered, this is a highly conservative assumption with extreme noise and drift rates. A lost aircraft with greater uncertainty searches a larger reference image or database (*27*), aided by the increased stability and distinctiveness of deep transformed imagery.

Training and test datasets

We train our transform using publicly available aerial orthoimagery from various regions of the United States (Fig. 3). Full-foliage ("leaf-on") and snow-cover imagery were obtained from the National Agricultural Imagery Program (NAIP) of the U.S. Department of Agriculture (*28*), and absent-foliage ("leaf-off") imagery was obtained from the geospatial data program of the state of Connecticut (*29*). Training and test sets were generated by coregistering cross-seasonal images into matching pairs. The datasets include man-made structures ranging from urban settlement to complete absence, with landcover including dense forest, agricultural fields, barren ground, coastline, and alpine tundra. We include a rugged mountainous dataset over the states of Wyoming and Montana, where classical registration performs poorly because of intense contrast changes caused by snowfall and severe mountain shadows. In addition to orthoimagery, we also test nonrigid transformations because of topography and off-nadir perspective by warping orthoimages onto coregistered digital elevation models (DEMs).

We consider two training/test dataset partition strategies, temporal and geographic, corresponding, respectively, to performance evaluation for VTRN and general registration use cases. Practical VTRN missions require a reference imagery database aboard the aircraft in advance and can always operate over their training dataset footprint. To evaluate the performance of our transform for VTRN, a representative test set is strictly temporally separated from the training set but overlaps with its geographic area. For general image registration, we consider the case in which the area of interest is too large to be practically contained in a single training set. Under either assumption, seasonal effects may have been only observed outside of the operating region (for example, extreme snowfall rarely appears in public datasets), so the ability to abstract structures beyond their specific geographic position is critical. Accordingly, we also partition the dataset into training and testing sets with strict geographic separation. For both cases, we assume a 4:1 ratio of training to test data, with only training data used to optimize the network.

Without the loss of generality, any of the domains can serve as a reference image at runtime depending on particular mission

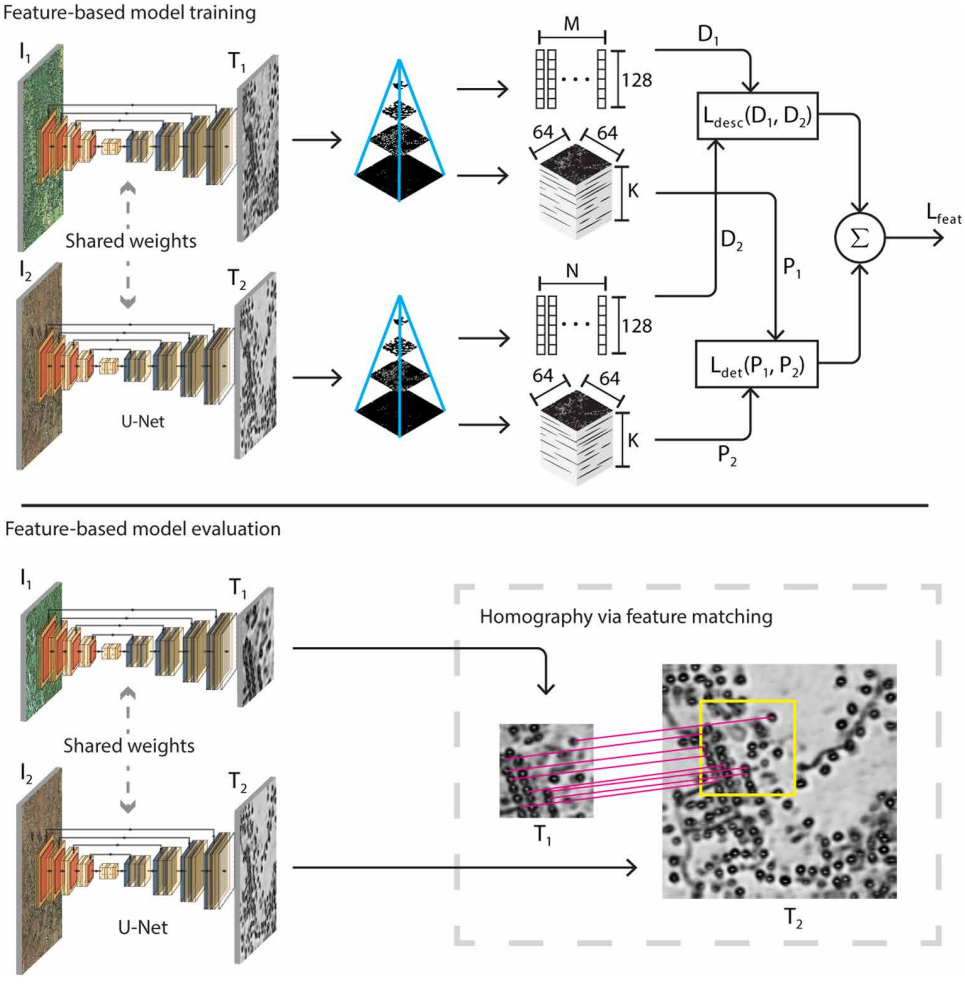

Fig. 2. Training and evaluation pipeline for feature-based image registration. During training, the loss function guides the network to transform images such that corresponding keypoints align in scale and location and their feature descriptors match. In evaluation mode, the network serves as a preprocessing step to transform images in different domains to the common domain in which feature-based registration excels.

requirements and the operating area. The map accuracy of each dataset is 6 m.

Leaf-drop dataset: Connecticut

The Connecticut set consists of 3638 coregistered leaf-on (summer 2016 and 2018) and leaf-off (early spring 2016) database image pairs randomly sampled over the state of Connecticut (Fig. 3A, left). Leaf-off database images were resampled to match the NAIP resolution data (0.6 m per pixel), with dimensions 1270 pixels by 1270 pixels. To evaluate performance when database images can be used in training, we use the 2016 leaf-on dataset collected 2 years before the training version (2018) for testing.

We also demonstrate a VTRN application with a simulated aircraft using 308 pairs taken over northwestern Connecticut. This imagery has an associated 10 cm DEM used to warp imagery and incorporate the effects of topography and off-nadir perspective. We note that this dataset is consistently more rugged and forested than the Connecticut set as a whole. The leaf-on images serve as NAVCAM imagery and are drawn from the same earlier edition used only for testing, and leaf-off images serve as a reference database.

Snowfall dataset: Rockies

The Rockies set is composed of 90 coregistered NAIP quarter-quadrangle image pairs, taken between 2012 and 2018, capturing summer and snow-cover conditions in the Rocky Mountains of Wyoming and Montana (Fig. 3A, right). For training and testing, we subdivide the quarter-quadrangles into 1200 pixels–by–1200 pixels nonoverlapping tiles with a resolution of 1 m per pixel. We note that this dataset is challenging even for manual registration because of extensive barren areas, a complete lack of man-made structures, snow and ice coverage, and severe mountain shadows. As with the Connecticut set, we also use earlier editions of the summer set solely for testing VTRN use cases.

Performance evaluation

After training, we evaluate transform performance by comparing the accuracy of widely used registration algorithms against a grayscale control. The test set consists of cross-seasonal pairs of source and reference images S and R that, respectively, produce registration queries and targets. We experimentally determine the best training procedure by restricting or combining different training sets and evaluating generality.

Because the typical failure mode of cross-seasonal registration is gross mismatch, we consider the correct match rate of image chips randomly drawn from each S and registered against the corresponding R as an estimate of the performance improvements afforded by a deep transform. This test is representative of typical VTRN operation in which a NAVCAM captures a small area within a database image, as well as nonrigid image registration in which translated chips are used to seed more complex transformations.

We use Intersection over Union (IoU) thresholds to identify match rates at varying levels of tolerance, as is standard for evaluating bounding box performance for detectors. If the IoU between a test chip and its predicted counterpart in the reference image is greater than the threshold, then the registration is counted as successful. If the IoU is less than the threshold, then it is counted as unsuccessful. We note that the limited map accuracy can prevent perfect IoU scores from being realized even with perfect matching.

For performance evaluation, we tested $K = 50$ randomly selected chips for each image pair in the test sets. As discussed above, we report match rate results for both temporally separated and geographically separated test sets.

Deep transforms for area-based registration

For area-based registration, we test a transform optimized for registration by NCC before discussing its relation to distribution-based

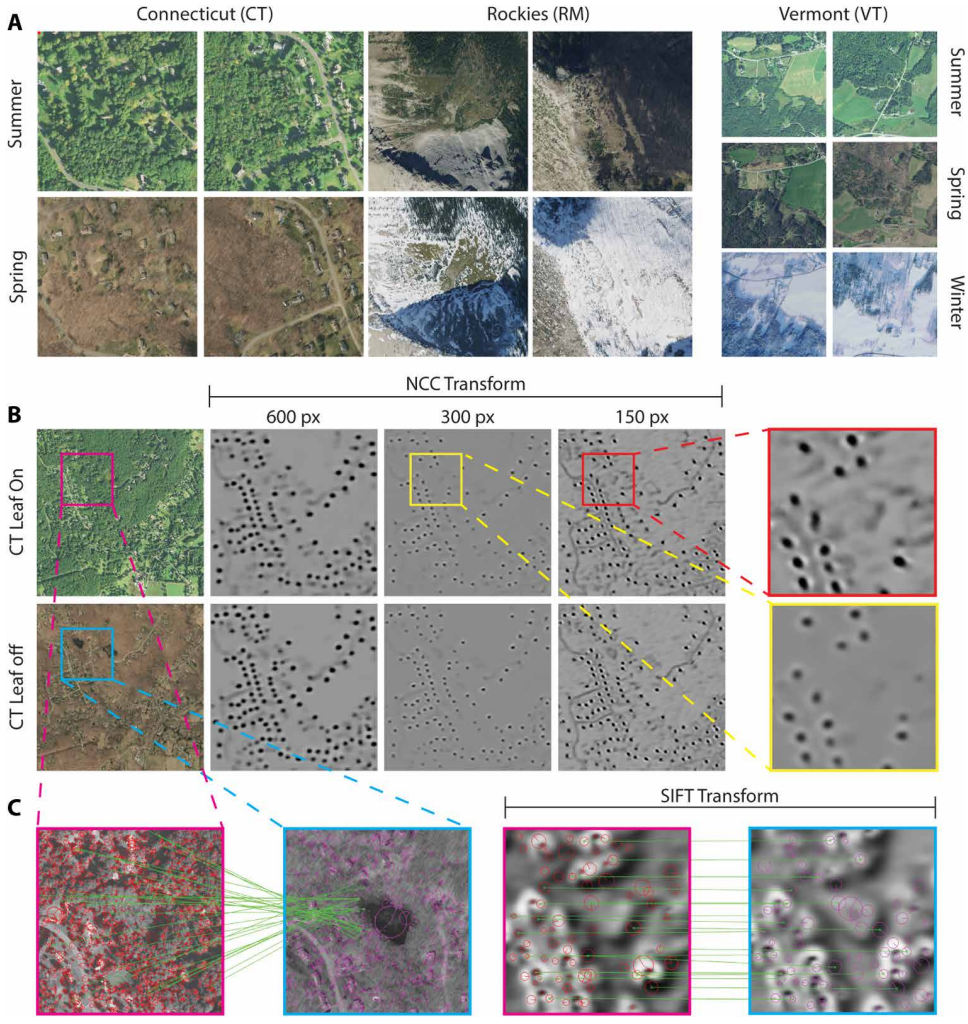

Fig. 3. Representative images from the datasets along with samples of NCC- and SIFT-based transforms. (**A**) Example images from the Connecticut (CT) and Rockies (RM) datasets in their various domains. (**B**) NCC transformations of the CT "leaf-on/leaf-off" image pair. The different effects of training using 600 pixels–by–600 pixels, 300 pixels–by–300 pixels, and 150 pixels–by–150 pixels image chips are shown. Zoomed-in images outlined in red and yellow highlight the details present when training with 150 pixels–by–150 pixels image chips. (**C**) The presence of a large number of incorrectly matched features in the grayscale image pair prevents RANSAC from finding the correct geometric transformation. The SIFT-optimized transform accentuates useful features for matching while stripping away noisy or unstable features that increase the number of outliers in matching.

methods. NCC is simply the linear correlation coefficient between two image patches, with a score of 1 if two images are perfectly positively correlated and a score of 0 if they are uncorrelated.

We first subject the training process to three sets of experiments to determine best practices: We consider the effect of training chip size on detail recovered by the transform, the volume of training data required for a transform to perform well on areas it has not been exposed to, and the volume of additional data required to perform well on areas with different landcover. Because each of these experiments consider the generality of the transform, we use the geographically partitioned test set.

Although performance at test time always improves with larger test chip sizes, we observed that a smaller training chip size produces better results. Test chips were fixed at 300 pixels on a side throughout our experiments, but training chips that were 150 pixels on a side outperformed larger chips, generated sharply localized structures, and enhanced detail in challenging areas such as uniform deciduous forests (Figs. 3 and 4).

Additional training data improve performance in a geographically separated test set but with returns that increase with IoU threshold because of improved sharpness (Fig. 4). Exposure to the full training dataset, rather than a randomly selected subset a 10th of its size, offers an 8 percentage-point improvement for the Connecticut set at an IoU threshold of 0.75, whereas increasing the IoU threshold to the range of 0.95 affords a 21 percentage-point improvement. Similarly, evaluation on the Rockies set yields a 4 percentage-point improvement at an IoU threshold of 0.75 but an 11 percentage-point improvement at a threshold of 0.95.

We also observe that additional training data with landcover different from the test set also improve matching rates (Fig. 4). On both the Rockies and Connecticut test sets, the best performance was achieved by training the network over all available Connecticut and Rockies training pairs rather than using each training set separately. For an IoU threshold of 0.9 and geographic partition of test and training sets, this network achieved match rates of 0.92 on the Connecticut set and 0.96 on the Rockies set compared with Connecticut-only and Rockies-only values of 0.83 and 0.94 and grayscale control values of 0.50 and 0.66. We note that the merged training set offered a greater improvement on the Connecticut test set than on the Rockies test set, particularly at IoU thresholds above 0.85. This asymmetry is largely due to the complete absence of man-made structures in the Rockies set. The Rockies set provides relevant training samples away from built-up areas in Connecticut, but Connecticut training samples cause the deep transform to rely on man-made structures that are irrelevant in wilderness environments. Expectedly, we also find that training sets must contain some landcover similar to the test set to offer increased performance over the grayscale control—training over Connecticut alone led to poor performance over the Rockies set and vice versa.

To consider VTRN use cases in which the flight area is known in advance, we merged the geographically separated training and test sets into a single training set and considered performance on additional leaf-on test sets separated temporally from the original sets (Fig. 4). Overall, we observe little impact from including the geographic area of the test area in training, with a small increase in

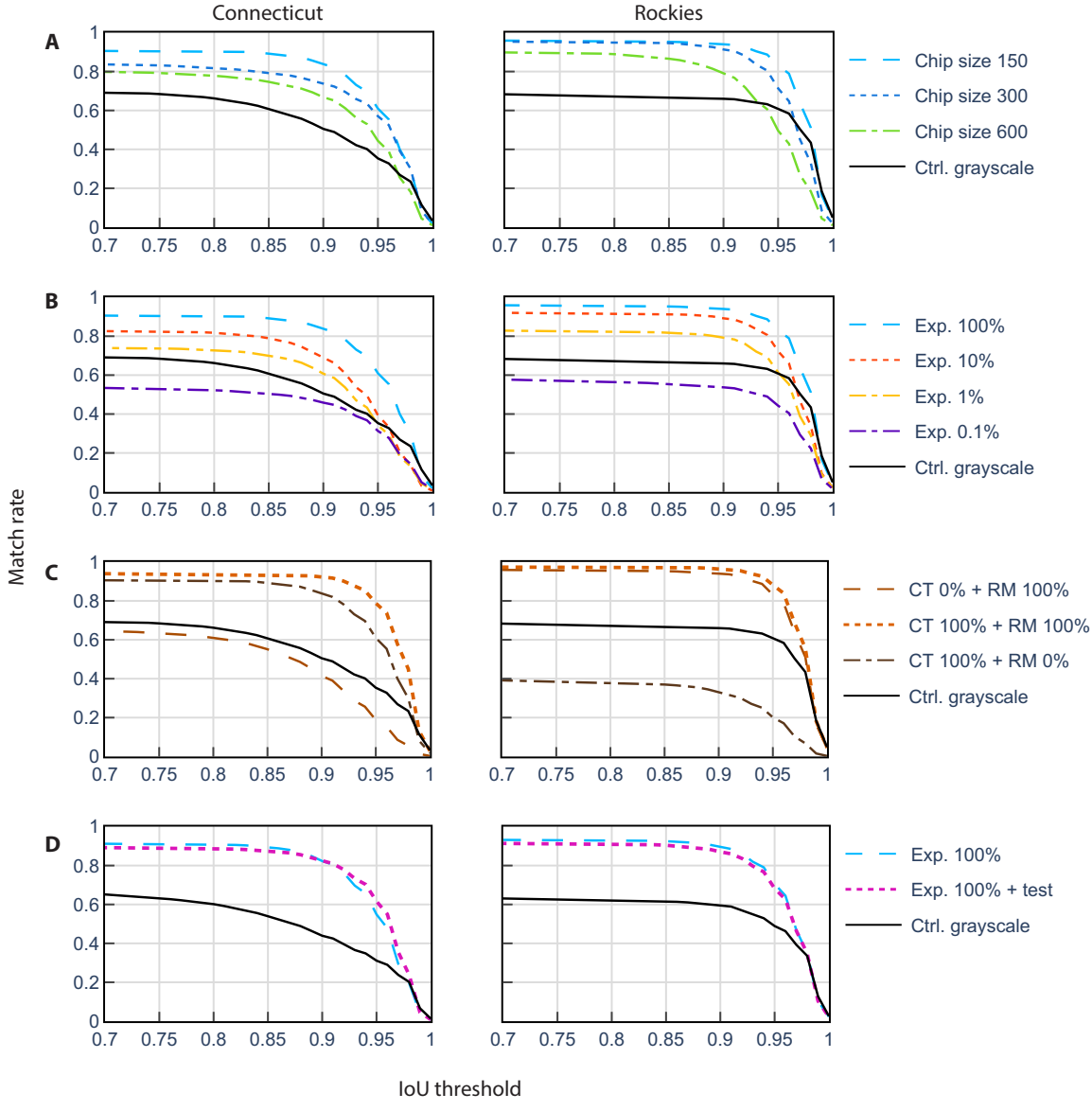

Fig. 4. NCC image registration results. Models in the first three rows were evaluated with geographically partitioned data, whereas those in the last row were tested with the temporally partitioned data. Unless otherwise noted, models were trained and tested using the datasets indicated by their column. (**A**) Plots in this row display the effect of varied chip sizes during training. (**B**) These plots show the positive match rate during testing while increasing the amount of training data seen by the transform model. (**C**) Transforms were trained on the entire CT set, the entire RM set, and a merged set consisting of both. Transforms were evaluated separately on the Connecticut and Rockies test sets. (**D**) Models trained specifically for VTRN perform on par with the best performing transforms from row (A).

performance at high IoU thresholds and a small decrease in performance at lower IoU thresholds. With the exception of those containing permanent structural changes, pairs that fail only when exposed to the additional data typically have intradomain landcover changes that were not covered in the training set. Because only one example of each domain was presented for each pair during training, this behavior is a symptom of overtraining a particular scene on a particular leaf-on appearance that can change severely at test time. On the other hand, the subtle increase at high IoUs appears to be due to increased sharpness afforded by inclusion of highly relevant training samples. The fact that the overall matching performance is relatively insensitive to geographic coverage, however, suggests that the features learned by the transform are abstract and high-level rather than tied to landmarks with a specific location.

We also attempted to train a transform to directly optimize a normalized MI objective but observed unstable training even with extremely small learning rates and deliberate overtraining over a single image pair also used for testing. This behavior is consistent with ill-posedness of MI as an optimization objective—the optimal transform is highly nonunique because MI and related distributional methods are invariant to any invertible deterministic function. Furthermore, MI is nondifferentiable and cannot be used for backpropagation without the use of a smooth approximation. Fortunately, the NCC training objective can also be used

to train transforms that improve MI-based registration. Although a full-resolution test of normalized MI performance over image chips is intractable [coarse-to-fine architectures are required even if heading and and altitude are known (*30*)], the NCC-trained transform enhances the ability of normalized MI to distinguish matching and nonmatching pairs. On a set of 11,600 matching and 11,600 nonmatching image chip pairs randomly drawn from the both test sets and exposed to our best-performing network, the Kullback-Leibler (KL) divergence between MI distributions for matching and mismatching grayscale image pairs was 0.22, whereas application of the NCC transform increases KL divergence to 1.58 (fig. S1).

Deep transforms for feature-based registration

Unlike the straightforward loss functions available for optimizing area methods, adapting our transform for feature-based registration performance requires simultaneous optimization of detector and descriptor response. We illustrate our approach using SIFT features (*10*) to which our transform adds seasonal invariance to well-known rotation, scale, and linear-brightness invariance properties. SIFT features fail spectacularly in grayscale control tests across seasonal pairs because most features detected are small and associated with unreliable ephemeral landscape textures such as vegetation. Instead, we simultaneously optimize a deep transform for detector reproduciblilty, descriptor reproducibility, and descriptor distinctiveness to remove unreliable seasonal content.

As for area-based methods, we evaluate the performance of our deep transform using geographically partitioned and temporally partitioned training and testing sets. To evaluate VTRN for a nadir-looking NAVCAM with no other sensors available (area methods assume a known orientation and height), we consider 640 pixels–by–480 pixels test chips extracted from cross-seasonal pairs and subjected to randomly varying translations, rotations, and scale transformations.

We observe that our deep transform offers major performance advantages over grayscale control across all IoU thresholds (Fig. 5) and over all test sets and training strategies considered. Feature-based methods appear more sensitive to the landcover encountered in training than area methods, however, and exposure to impertinent samples can degrade performance. Merging the Rockies and Connecticut datasets improved match rates on the Connecticut test set at all IoU thresholds, largely due to improved feature density in forested areas, but worsened match rates on the Rockies test set because of a reliance on man-made structures that are not present at test time.

In addition to improving match rates, the deep transform also provides a realistic assessment of the navigational utility of an image and fails far more gracefully than grayscale imagery. Images devoid of navigational cues, such as open water or uniform deciduous forest, simply do not have enough features available for registration after exposure to the transform and result in a failure. Grayscale imagery, on the other hand, often has enough spurious features available to return a registration estimate and tends to generate mismatches when it should instead fail. Accordingly, we report experiments that both reject and retain failed test pairs in Fig. 5.

Last, including the area of a temporally separated test set within the training set slightly reduces the number of rejected pairs on the Rockies set but with essentially no improvement on the Connecticut test set and a slight decrease in matching skill on accepted pairs in both sets. This decrease in performance is consistent with overtraining on the leaf-on appearance of the scene in training.

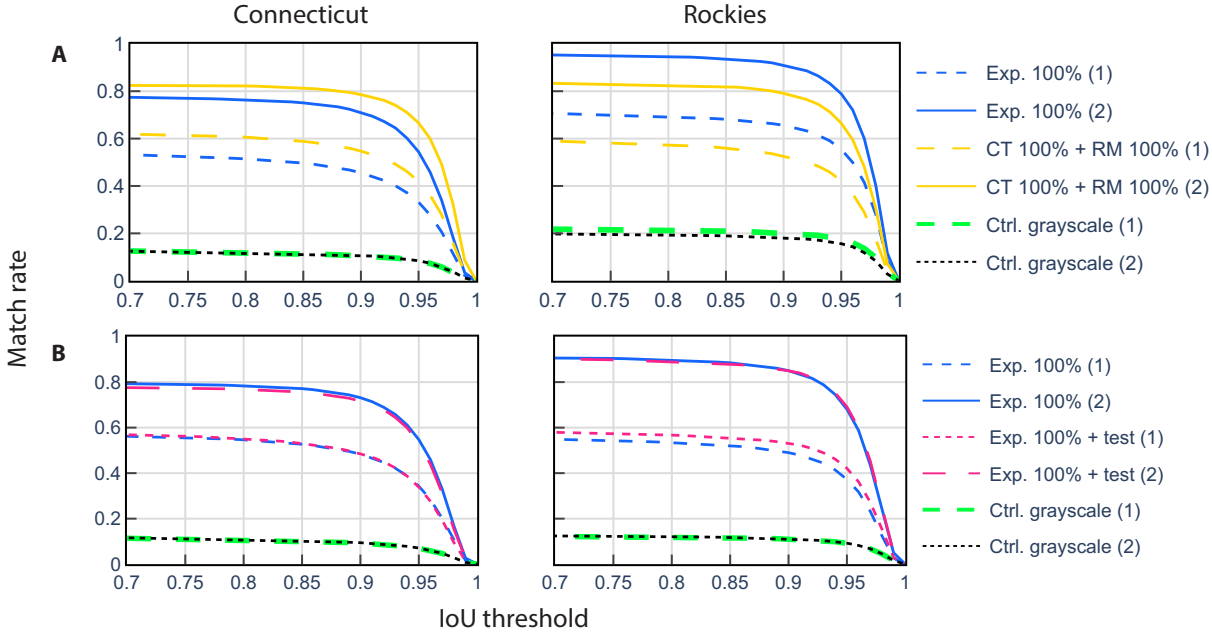

Fig. 5. SIFT matching performance on geographically and temporally separated test sets evaluated using 640 pixels-by-480 pixels test image chips. Test pairs that lack sufficient keypoints to calculate a camera pose are counted as mismatches in experiments denoted "(1)" but assumed useless for navigation in experiments denoted "(2)" and not counted. (**A**) For geographically separated evaluations, we compare the matching rates of transformations trained on a single Connecticut (CT) or Rockies (RM) set ("Exp. 100%", for CT or RM) against transformations trained on both sets ("CT 100% + RM 100%"). (**B**) To determine the effect of training over an anticipated flight area, we compare transformations trained on geographically separated data ("Exp. 100%") to transformations specifically trained over the same area as the test set but from different years ("Exp. 100% + Test").

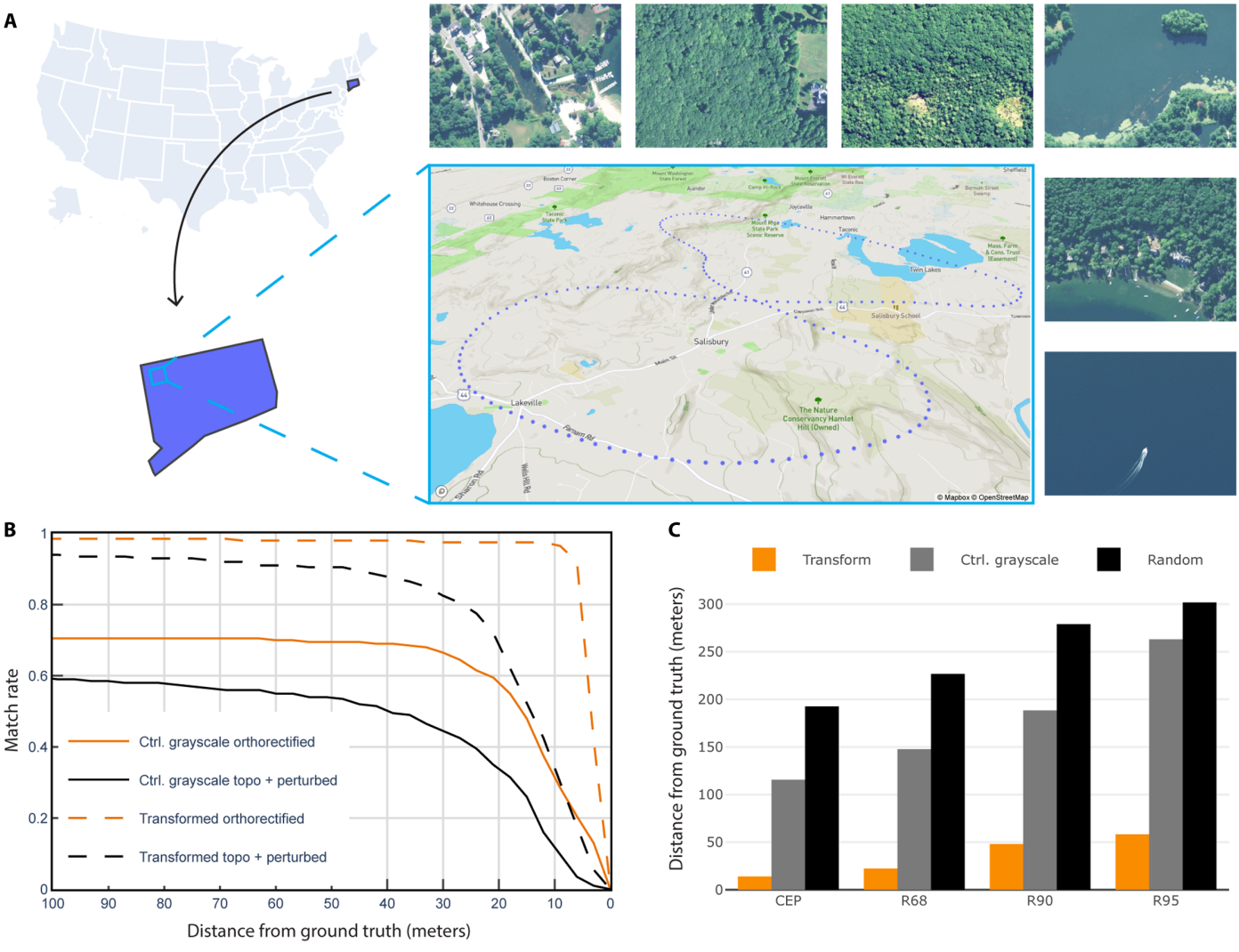

Fig. 6. VTRN demonstration with a simulated flight over northwestern Connecticut. (**A**) The figure-eight flight trajectory is shown with a few select NAVCAM images (640 pixels by 480 pixels) seen by the aircraft. The trajectory contains a mixture of small towns, agricultural areas, dense deciduous forest, and occasional open water. (**B**) NCC-based registration match rate at various distance thresholds from the ground truth positions. (**C**) SIFT-based registration distance-from-ground truths at the 50th, 68th, 90th, and 95th percentiles.

Seasonally invariant VTRN demonstration

We evaluate our transform under realistic VTRN conditions during a simulated flight over a relatively undeveloped and rugged area of northwest Connecticut (Fig. 6). The conditions encountered during the flight are considerably more challenging overall than the Connecticut set and contain large uninterrupted expanses of deciduous forest with steeper terrain and sparser development. The flight area is covered by a DEM and a temporally separated test set, which are used to simulate imagery from a NAVCAM aboard a fixed-wing aircraft. As a result of the steep terrain and rolling motion of the aircraft, image registration involves a complex geometric transformation incorporating aircraft translation, attitude, and height changes along with and perspective effects because of a combination of camera optics, off-nadir viewing, and topography. Reference imagery is drawn from a database of leaf-off orthophotography, with the NAVCAM imagery taken in full leaf-on conditions separated by 2 years from the training data. Because of the complexity of this transformation and the intense seasonal changes, both the content invariance of deep transforms and the geometric advantages of conventional registration are needed to generate reliable absolute position updates.

We consider both NCC- and SIFT-based navigation along with associated deep transforms and a grayscale control set. SIFT can accommodate complex geometric transforms, in principle, as long as the number of features matched is adequate, whereas NCC assumes nadir-looking shots and an estimate of heading provided by a compass and an estimate of height provided by an altimeter. We do not supply SIFT with a heading estimate because of its rotational invariance, but NCC is supplied with perturbed (±2°) on-nadir shots and noisy heading estimates (±5°) to evaluate realistic operating conditions with instrument error. Contrary to typical practice, we also evaluate NCC without the benefit of online orthorectification. To isolate seasonal effects on performance from geometric effects, we also include a control NCC test

with online orthorectification, perfect altimetry, and perfect compass measurements.

The test NAVCAM sequence consists of 200 image chips with dimensions 640 pixels by 480 pixels, taken approximately every 180 m along a 36-km figure-eight aircraft trajectory about 200 m above the terrain. Because of the frequent and sequential nature of the imagery, shots are somewhat correlated and cluster in challenging areas that were far less abundant in the full Connecticut set. We use a database image of 1270 pixels by 1270 pixels for NCC and a smaller 800 pixel–by–800 pixel image for SIFT because of its sparser nature and use with poorly constrained transformations. We also note that 3 of the 200 shots contained only open water and were either mismatched or rejected for lacking navigational content by the registration algorithms in all experiments.

For NCC, the VTRN backend was performed with a patch-based sliding window method, whereas SIFT features were matched using standard brute-force sum-of-absolute-differences scores. Extremely noisy onboard odometry was simulated and used to select reference images (0.6 m per pixel) that contain the aircraft location somewhere within the prescribed map size. This odometry error assumption far exceeds the average interval between shots of 180 m. Each shot was treated as independent, and odometry was not allowed to influence the initial pre-VTRN uncertainty beyond the size of the reference image.

Test images were subjected to the best-performing NCC and SIFT transforms determined in performance evaluation for feature-based and area-based methods, which were trained over Connecticut and Rockies training data consolidated into a single set. SIFT feature detection, extraction, and matching were performed using OpenCV with default settings, and random sample consensus (RANSAC) was used to estimate a full-affine geometric transformation without knowledge of the aircraft height or attitude. Off-nadir perspective and topography allow test images to differ in shape from ground truth and prevent perfect IoU scores even if localization is perfect, so navigation performance was evaluated using the centroid distance between registered images and ground truth. We also calculate several centroid distance statistics, including circular error probable (CEP), R68, R90, and R95 scores, which correspond, respectively, to the 50th, 68th, 90th, and 95th percentiles of centroid distance.

VTRN performance

When used with a deep transform, NCC proved to be a powerful and robust navigation tool, even in the presence of steep topography and viewing angle perturbations that violated its geometric assumptions. The deep transform had far fewer mismatches than grayscale and also localized close matches more accurately (Fig. 6). The CEP for the unmodified transformed image was 14 m compared with 3 m for the idealized geometry, 40 m for unmodified grayscale, 16 m for the idealized grayscale geometry, and 168 m for a set of 100 randomly guessed centroids per frame. The deep transform had few gross mismatches, with transformed imagery having an R68 score of 19 m, an R90 score of 46 m, and an R95 score of 115 m compared with 4, 7, and 14 m for idealized conditions. Because each of these scores are considerably smaller than the size of the NAVCAM image even at high percentiles, this divergence suggests that much of the error was caused by the impossibility of matching distorted test images against rectangular, orthorectified ground truth using a rigid translation. Grayscale control imagery, on the other hand, had an R68 score of 150 m, an R90 score of 233 m, and R95 score of 262 m, which improved to 32, 228, and 260 m under ideal imaging geometries. The departure in R68 but approximate convergence in R90 and R95 score between these two experiments suggests that grayscale was highly sensitive to topography and viewing angle. Furthermore, grayscale was also plagued by gross mismatches caused by seasonal content, as evidenced by comparison with random centroid guessing scores of 197, 241, and 258 m.

SIFT was able to provide accurate and reliable accurate position estimates from deep transformed imagery even with a camera as the sole navigation instrument but at the price of a much lower update rate. Although the deep transform was able to stabilize imagery, it struggled to enhance and concentrate adequate numbers of strong features in the most challenging areas. The stabilizing effect, however, recovered high levels of accuracy among useful images identified using an empirical feature match count. Eleven percent of the images were accepted as reliable in the transformed imagery, with a CEP of 14 m and a maximum error of 76 m. In contrast, grayscale control imagery was unable to anticipate reliable or unreliable frames at any feature count threshold. For the same experiment, 53% of the images were accepted, with a CEP of 116 m and an R95 of 263 m for a minor advantage over random guess values of 192 and 301 m, respectively.

Hardware and runtime metrics

Our machines were configured with Intel Core i9-7900x processors with 128 gigabyte (GB) of memory. We trained the deep transforms using Nvidia Quadro RTX 8000 (48 GB) and the Titan RTX (24 GB) graphics processing units (GPUs). Because VTRN requires image chips to be transformed in an online setting, we recorded runtime and memory usage for various navigation chip sizes, including those used in our VTRN demonstration. We benchmarked the deep transform using the Titan RTX and report these results in Table 1. We find that the GPU-accelerated deep transform can process 640 pixel–by–480 pixel chips at 17 Hz and even larger 1600 pixel–by–1200 pixel chips at 2.7 Hz. For small image chips, the deep transform can still operate at reasonable speeds even without a GPU but at a cost of a slower position update rate. We do not report metrics for larger database images, which are not transformed in real time but instead calculated offline in advance and cached for quick lookup.

DISCUSSION

Overall, our transform learns highly general, abstract features that are semantic in nature rather than landmarks tied to specific geographic locations. Our transform need not be trained over the exact

Table 1. Runtime and memory consumption of the deep transform for various input image sizes.

Image size (pixels by pixels)	GPU I/O (s)	GPU transform (s)	GPU memory (GB)	CPU transform (s)
300 × 300	0.0003	0.0192	0.67	0.700
640 × 480	0.0004	0.0597	2.13	2.734
1280 × 720	0.0008	0.1722	6.25	8.139
1600 × 1200	0.0015	0.3623	12.95	17.250

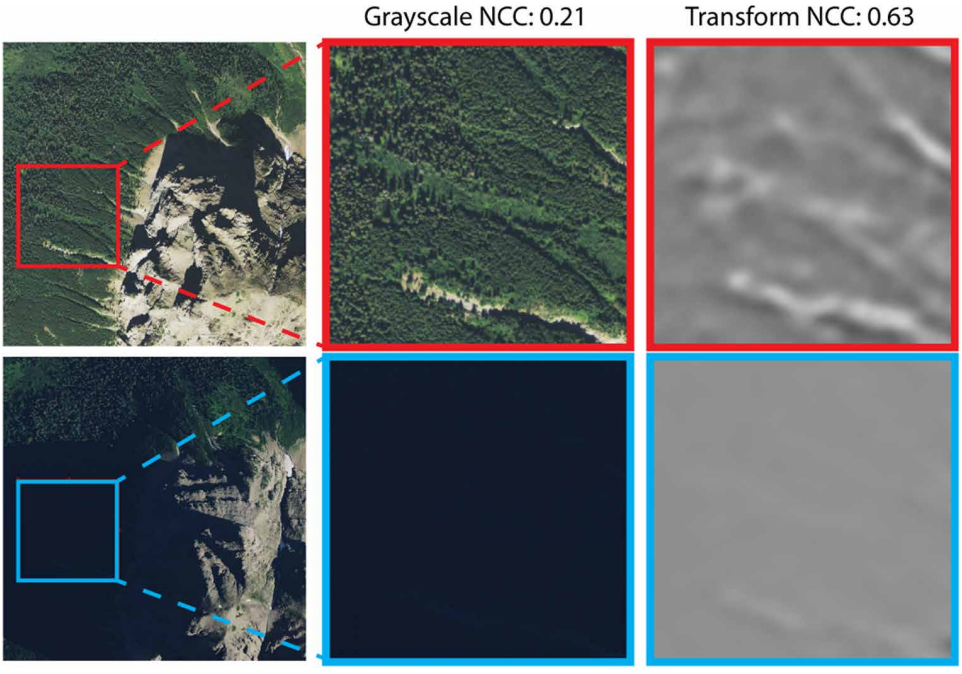

Fig. 7. Example of improved robustness to severe mountain shadows in test data. Self-supervised training causes the transform to attempt to mitigate any unstable content without explicit annotation. Intense lighting changes, for example, were occasionally present in the Rockies dataset and were addressed by the transform, although they were not specifically identified as a relevant seasonal disturbance. Despite information loss to saturation, areas in deep shadow were brightened (left, red and blue boxes are coregistered), and stable landscape features were enhanced. The transform was not exposed to either image during training.

area where it will be used, and only small amounts of data were required to surpass performance on standard grayscale imagery. The advantages of isolating geometry to an engineered conventional backend, rather than training, were also apparent in feature-based experiments. Using a SIFT backend, deep transforms were able to accurately assess reliability and accommodate complex topographic and perspective transformations despite training only on orthophotography. We also find that NCC is an extraordinarily powerful tool when supplemented with a deep transform despite its simplicity. Deep-transformed NCC exhibited robust performance against severe seasonal effects and perturbed imaging conditions that violated its translation-only, mutually orthorectified geometric assumptions.

The usual case of more data improving performance remains largely true, but targeted training set design can help achieve better results. Training sets must include at least some landcover similar to what will be encountered at test time to be effective, with increasing volumes improving the spatial precision of the transform. Including as much data as possible improved matching for area-based methods even if additional training examples had highly dissimilar landcover, but feature-based approaches required care to avoid crippling a deep transform with irrelevant samples. Although the inclusion of roadless areas from the Rockies training set improved feature matching away from built-up areas in Connecticut, for example, overexposure to buildings in training also hindered the ability of the deep transform to construct useful SIFT features in images where they were not present. Consequently, systems designed to operate entirely away from man-made structures, such as for extraterrestrial applications, should not use training sets dominated by them. Our transform also accommodated ephemeral changes that were not anticipated in advance, which suggests that general time-series data are useful for stabilizing VTRN even if seasonal disturbances are not expected. Deep mountain shadows that change throughout the day and year, for example, had improved matching as a natural consequence of their presence in the set and the self-supervised architecture (Fig. 7). This tendency is further evidenced by the possibility of overtraining over specific leaf-on appearances, which caused inclusion of VTRN database imagery in training to decrease performance on a temporally separated test set in some cases.

Although feature-based methods using SIFT features are more flexible in the geometric transformations that they can accommodate, the NCC training objective was much more robust in the most challenging and unstable areas. Unless operational considerations preclude the use of NCC and require feature-based approaches, such as the presence of extreme topography at low altitudes or the lack of an attitude estimate, the additional information afforded by NCC-based matching is preferable to the geometric flexibility of SIFT when severe content changes are present. NCC was able to infer reproducible structure in aggressively changing environments where SIFT had difficulty finding features and also proved to be robust to local topographic and viewpoint perturbations that would ordinarily call for the use of feature-based approaches. Nonetheless, the deep transform enhanced the reliability of SIFT-based navigation in areas that were impossible for grayscale imagery—unusable shots simply lacked features and were consistently identified in advance. Unlike grayscale, which proposed overconfident matches little better than random guessing, unusable shots were discarded rather than being naively allowed to disrupt navigation.

The NCC objective also supported registration by MI maximization without incurring its extreme computational expense or the complexity of backpropagation through a nondifferentiable objective (*31*). NCC is well known to be equivalent to MI for normally distributed random variables, and transforms trained on NCC improved the separation of matching and nonmatching MI scores (fig. S1) while also being much easier and faster to train. Because such transforms are already directly optimized for NCC, MI-based registration architectures and their inherent difficulties can be replaced entirely by a real-time NCC-based architecture with a deep transform.

Last, the results presented here isolate difficulties associated with content changes and consider odometry and filtering only to the extent that they generate reference images including the aircraft location. Filtering can serve not only to improve running estimates of position, as with any navigation technique, but also to aid the VTRN matching process itself by restricting the size of a registration

problem to a tight uncertainty. If position and attitude are sufficiently well known, then a registration solution can be proposed in advance and VTRN used to modify and confirm it within tight bounds. Feature-based approaches stand much to gain from filtering in particular, because the relatively unconstrained nature of the geometric transformations they solve benefits highly from reliable initial guesses and bounds.

MATERIALS AND METHODS

We structure our transform as an FCN, with a self-supervised training architecture that assumes coregistered image pairs but identifies and extracts seasonally stable structures on its own. During training (Fig. 2), image pairs with cross-seasonal differences (leaf-on versus leaf-off or summer versus snow) are consecutively exposed to the FCN with shared weights, which outputs a pair of single-channel transformed images with the same dimensions as the inputs. The network is trained using a loss function calculated from the performance of a specified registration technique (for example, NCC or SIFT) on an auxiliary chip-matching task. Although the specific structure of the loss function depends on the choice of registration backend, the transform can be trained to optimize any algorithm with a differentiable and well-posed matching score. In doing so, we harness the statistical power of deep learning to add seasonal robustness to legacy techniques.

After training, the FCN constitutes a single-stream image transform that strips source and reference images of their seasonally unstable content for stable registration. In the rest of this section, we highlight specific network details and consider the construction of seasonally invariant transforms for area-based and feature-based registration.

Network architecture

We chose U-Net (*26*) as an example model, although any FCN architecture that preserves image resolution will suffice. In a deep transform context, U-Net maps an input grayscale image of dimensions W pixels by H pixels, with intensity values in [0,1], to an output grayscale image of the same size and compressed to [0,1] using a sigmoid function. Inputs are normalized using a fixed mean and SD derived from the training set. To avoid "checkerboard" artifacts associated with deconvolution, we replace the deconvolution layers of the original U-Net with blocks consisting of an upsampling operation with bicubic interpolation followed by a convolution (*32*).

Loss function and regularization

During training, the loss function is calculated by passing pairs of transformed chips (T_i^1, T_i^2) to the matching function of a desired image registration algorithm. The matching function generates a normalized similarity score $\hat{y}(T_i^1, T_i^2) \in [0,1]$. The normalized similarity score $\hat{y}$ is in turn compared with a binary label y_i that indicates whether the chips were originally coregistered ($y_i = 1$) or not ($y_i = 0$). The loss $\mathcal{L}$ is the sum of the contributions from each image pair and is used to update the network weights by backpropagation. Accordingly, the chip pair (T_i^1, T_i^2) contributes

$$\mathcal{L}^{(i)} = \| \hat{y}(T_i^1, T_i^2) - y_i \| \tag{1}$$

to $\mathcal{L}$, where $\|\cdot\|$ is a differentiable norm. The only requirement for $\hat{y}$ is that the matching score be differentiable with respect to the network parameters and well posed for backpropagation.

To avoid trivial transforms, such as setting all pixels identically to zero, nonmatching samples with $y_i = 0$ must be presented during training. Coregistered (positive) chips are selected with probability 0.5 and driven toward a perfect matching score, whereas nonmatching (negative) samples are driven toward either a worst-case mismatch score or separation margin.

Implementation and training details

Our U-Net architecture is based off of the GitHub repository, Pytorch-UNet (*33*), and adapted by replacing the deconvolutions to resolve checkerboard artifacts as mentioned before. All transforms were trained using the Adam optimizer with a learning rate of 1×10^{-5} over 300 epochs. The learning rate was decayed at an exponential rate of 0.995 every two epochs. Batch size was adjusted depending on training chip size to fit on the GPU, but no larger than 16. We relied on the OpenCV and Kornia Python libraries to implement the loss function for SIFT-based image registration (*34*, *35*).

Area methods

Our training procedure for area-based registration methods directly optimizes registration performance on the training set. All area-based registration techniques use a similarity score based on intensity values to determine whether image chips match or not, which is simply driven toward perfect (matching) or worst-case (nonmatching) values to train our network. The NCC objective is widely applicable and also useful for MI and related area-based methods.

Normalized cross-correlation

For a chip pair (T_i^1, T_i^2), the zero-mean NCC score is defined as

$$\hat{y}_i^{\mathrm{NCC}} = \frac{1}{n\sigma_u\sigma_v}\sum_{u,v}(T_i^1(u,v)-\mu_1)(T_i^2(u,v)-\mu_2) \tag{2}$$

where u, v are pixel coordinates in the chips, n is the number of pixels in each chip, and μ_1, σ_1 and μ_2, σ_2 are the respective means and SDs of the transformed chips. To optimize the transform, we drive the similarity score $\hat{y}_i^{\mathrm{NCC}}$ toward the label $y_i = 1$ for positive samples and $y_i = 0$ for negative samples using squared error over K pairs

$$\mathcal{L}_{\mathrm{NCC}} = \frac{1}{K}\sum_{i=0}^{K} \| \hat{y}_i^{\mathrm{NCC}} - y_i \|_2^2 \tag{3}$$

Feature methods

Unlike area methods, feature-based methods consist of two objectives that must be optimized: detection in which reliable feature locations are identified and extraction in which descriptors that consistently represent unique structures are selected at detected locations (Fig. 2).

Scale-invariant feature transform

To optimize for SIFT, we incorporate the two objectives mentioned above into our loss function. Detector response is driven to a common domain using NCC optimization over the difference-of-Gaussian (DoG) pyramids from a pair of transformed image chips (T_i^1, T_i^2)

$$\mathcal{L}_{\mathrm{det}} = \frac{1}{K}\sum_{i=0}^{K} \| \hat{y}^{\mathrm{NCC}}(P_j^1, P_j^2) - y_i \|_2^2 \tag{4}$$

where (P_j^1, P_j^2) are pairs of 64×64 patches sampled from the DoG pyramids of the transformed input pair. During training, we randomly extract 100 pyramid sample pairs from each input image with a negative matching rate of 0.5.

To optimize descriptor performance, we calculate the pairwise distance between the M detected keypoints in T_i^1 and the N detected

keypoints in T_i^2 (Eq. 5). During training, M and N are each fixed at 500. Normalized descriptor pairs $\left(D_m^1, D_n^2\right)$ are extracted from each image, where D_m^1 is the mth descriptor extracted from T_i^1 and D_n^2 is the nth descriptor extracted from T_i^2. The Euclidean distance between the descriptor pairs $\hat{y}_{m,n}^{\text{desc}}$ is driven toward a label $y_{m,n}$, which is 0 if the corresponding keypoints match in scale and location, and driven toward the maximum margin $a = 2$ if they do not

$$\hat{y}_{m,n}^{\text{desc}} = \|D_m^1 - D_n^2\|_2 \tag{5}$$

$$\mathcal{L}_{\text{desc}} = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N} y_{m,n}\cdot\hat{y}_{m,n}^{\text{desc}} + (1 - y_{m,n})\cdot\max(0, a - \hat{y}_{m,n}^{\text{desc}}) \tag{6}$$

The two loss functions are then jointly optimized as

$$\mathcal{L} = \beta\mathcal{L}_{\text{det}} + \mathcal{L}_{\text{desc}} \tag{7}$$

where β is empirically chosen to be 10. To handle more keypoints at different scales, we randomly and identically scale the input pairs between 0.6 and 1.4 and crop to 256 pixels by 256 pixels, 400 pixels by 400 pixels, or 512 pixels by 512 pixels before transformation.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/6/55/eabf3320/DC1
Fig. S1
Movies S1 to S3

REFERENCES AND NOTES

1. A. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, A. Ansar, L. Matthies, Vision-aided inertial navigation for spacecraft entry, descent, and landing. *IEEE Trans. Robot.* **25**, 264–280 (2009).
2. L. Downes, T. J. Steiner, J. P. How, Deep learning crater detection for lunar terrain relative navigation, in *American Institute of Aeronautics and Astronautics Scitech 2020 Forum* (American Institute of Aeronautics and Astronautics, 2020).
3. J. R. Carr, J. S. Sobek, Digital Scene Matching Area Correlator (DSMAC), in *Image Processing for Missile Guidance*, T. F. Wiener, Ed. (SPIE, 1980).
4. G. Conte, P. Doherty, Vision-based unmanned aerial vehicle navigation using geo-referenced information. *EURASIP J. Adv. Signal Process* **2009**, 387308 (2009).
5. J. Townshend, C. Justice, C. Gurney, J. McManus, The impact of misregistration on change detection. *IEEE Trans. Geosci. Remote Sens.* **30**, 1054–1060 (1992).
6. J. L. Moigne, N. S. Netanyahu, R. D. Eastman, *Image Registration for Remote Sensing* (Cambridge Univ. Press, 2009).
7. W. K. Pratt, *Digital Image Processing: PIKS Scientific Inside* (Wiley-Interscience, 2007).
8. P. Viola, W. M. Wells III, Alignment by maximization of mutual information. *Int. J. Comput. Vis.* **24**, 137–154 (1997).
9. B. Reddy, B. Chatterji, An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. Image Process.* **5**, 1266–1271 (1996).
10. D. G. Lowe, Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**, 91–110 (2004).
11. N. Longbotham, F. Pacifici, S. Malitz, W. Baugh, G. Camps-Valls, *Fourier transform spectroscopy and hyperspectral imaging and sounding of the environment* (OSA, 2015).
12. J.-P. Avouac, S. Leprince, Geodetic imaging using optical systems, in *Treatise on Geophysics* (Elsevier, 2015), pp. 387–424.
13. J. P. Golden, Terrain contour matching (TERCOM): A Cruise missile guidance aid, in *Image Processing for Missile Guidance*, T. F. Wiener, Ed. (SPIE, 1980).
14. F. Kendoul, Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *J. Field Robot.* **29**, 315–378 (2012).
15. M. D. Zeiler, R. Fergus, Visualizing and Understanding Convolutional Networks, in *Computer Vision* (ECCV, 2014), pp. 818–833.
16. G. Haskins, U. Kruger, P. Yan, Deep learning in medical image registration: A survey. *Mach. Vis. Appl.* **31**, 8 (2020).
17. S. Zagoruyko, N. Komodakis, Learning to compare image patches via convolutional neural networks, in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2015), pp. 4353–4361.
18. L. H. Hughes, M. Schmitt, L. Mou, Y. Wang, X. X. Zhu, Identifying corresponding patches in SAR and optical images with a pseudo-Siamese CNN. *IEEE Geosci. Remote Sens. Lett.* **15**, 784–788 (2018).
19. M. Blendowski, M. P. Heinrich, Combining MRF-based deformable registration and deep binary 3D-CNN descriptors for large lung motion estimation in COPD patients. *Int. J. Comput. Assist. Radiol. Surg.* **14**, 43–52 (2019).
20. Z. Yang, T. Dan, Y. Yang, Multi-temporal remote sensing image registration using deep convolutional features. *IEEE Access* **6**, 38544–38555 (2018).
21. A. Gupta, Y. Peng, S. Watson, H. Yin, Multitemporal aerial image registration using semantic features, in *Intelligent Data Engineering and Automated Learning* (IDEAL, 2019), pp. 78–86.
22. A. Nassar, K. Amer, R. ElHakim, M. ElHelw, A deep CNN-based framework for enhanced aerial imagery registration with applications to uav geolocalization, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2018), pp. 1594–159410.
23. D. Tuia, C. Persello, L. Bruzzone, Domain adaptation for the classification of remote sensing data: An overview of recent advances. *IEEE Trans. Geosci. Remote Sens.* **4**, 41–57 (2016).
24. Z. Murez, S. Kolouri, D. Kriegman, R. Ramamoorthi, K. Kim, Image to image translation for domain adaptation, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 4500–4509.
25. A. Anoosheh, T. Sattler, R. Timofte, M. Pollefeys, L. V. Gool, Night-to-day image translation for retrieval-based localization, in *2019 International Conference on Robotics and Automation (ICRA)* (2019), pp. 5958–5964.
26. O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, 2015), pp. 234–241.
27. R. Arandjelovic, A. Zisserman, All about VLAD, in *2013 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2013), pp. 1578–1585.
28. U.S. Department of Agriculture Farm Service Agency, Aerial Photography Field Office, National Agricultural Imagery Program (2012–2019); earthexplorer.usgs.gov.
29. Capitol Region Council of Governments of Connecticut, CRCOG Orthoimagery (2016); https://cteco.uconn.edu/data/flight2016/.
30. A. Ansar, L. Matthies, Multi-modal image registration for localization in titans atmosphere, in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2009), pp. 3349–3354.
31. M. Unser, P. Thevenaz, Optimization of mutual information for multiresolution image registration. *IEEE Trans. Image Processing* **9**, 2083–2099 (2000).
32. A. Odena, V. Dumoulin, C. Olah, Deconvolution and checkerboard artifacts (Distill, 2016); http://distill.pub/2016/deconv-checkerboard/.
33. A. Milesi, Pytorch-Unet (2017); https://github.com/milesial/Pytorch-UNet.
34. G. Bradski, The OpenCV library. *Dr. Dobb's J. Software Tools* **25**, 120–125 (2000).
35. E. Riba, D. Mishkin, D. Ponsa, E. Rublee, G. Bradski, Kornia: An Open source differentiable computer vision library for PyTorch, in *2020 IEEE Winter Conference on Applications of Computer Vision* (IEEE, 2020), pp. 3663–3672.
36. H. Iqbal, PlotNeuralNet (2018); https://doi.org/10.5281/zenodo.2526396.

Acknowledgments: We thank P. Tokumaru. **Funding:** This project was in part funded by the Boeing Company with R. K. Li as Boeing Project Manager. C.T.L. acknowledges the National Science Foundation Graduate Research Fellowship under grant no. DGE1745301. A.S.M. was in part supported by Caltech's Summer Undergraduate Research Fellowship (SURF). **Author contributions:** A.T.F. developed the deep transform, implemented the software, designed the test and training datasets, and designed the experiments. C.T.L. contributed to architecture development, implemented the software, and conducted the experiments. A.S.M. contributed to the VTRN demonstration and conducted the experiments. S.-J. C. contributed to development of the VTRN concept and robotics applications and directed the research activities. All authors participated in the preparation of the manuscript. **Competing interests:** A.T.F., C.T.L., and S.-J. C. are inventors on a pending patent submitted by the California Institute of Technology that covers the material described herein. The authors declare that they have no other competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials. Imagery used and the U-Net network used are publicly available at the relevant cited sources.

Submitted 19 October 2020
Accepted 1 June 2021
Published 23 June 2021
10.1126/scirobotics.abf3320

Citation: A. T. Fragoso, C. T. Lee, A. S. McCoy, S.-J. Chung, A seasonally invariant deep transform for visual terrain-relative navigation. *Sci. Robot.* **6**, eabf3320 (2021).

Science Robotics

A seasonally invariant deep transform for visual terrain-relative navigation

Anthony T. Fragoso, Connor T. Lee, Austin S. McCoy, and Soon-Jo Chung

Sci. Robot. **6** (55), eabf3320. DOI: 10.1126/scirobotics.abf3320

Science Robotics (ISSN 2470-9476) is published by the American Association for the Advancement of Science. 1200 New York Avenue NW, Washington, DC 20005. The title *Science Robotics* is a registered trademark of AAAS.