MotionTransformer: Transferring Neural Inertial Tracking between Domains

Changhao Chen,[1] **Yishu Miao,**[1] **Chris Xiaoxuan Lu,**[1] **Linhai Xie,**[1]
Phil Blunsom,[1,2] **Andrew Markham,**[1] **Niki Trigoni**[1]
[1]Department of Computer Science, University of Oxford
[2]DeepMind
firstname.lastname@cs.ox.ac.uk

Abstract

Inertial information processing plays a pivotal role in egomotion awareness for mobile agents, as inertial measurements are entirely egocentric and not environment dependent. However, they are affected greatly by changes in sensor placement/orientation or motion dynamics, and it is infeasible to collect labelled data from every domain. To overcome the challenges of domain adaptation on long sensory sequences, we propose MotionTransformer - a novel framework that extracts domain-invariant features of raw sequences from arbitrary domains, and transforms to new domains without any paired data. Through the experiments, we demonstrate that it is able to efficiently and effectively convert the raw sequence from a new unlabelled target domain into an accurate inertial trajectory, benefiting from the motion knowledge transferred from the labelled source domain. We also conduct real-world experiments to show our framework can reconstruct physically meaningful trajectories from raw IMU measurements obtained with a standard mobile phone in various attachments.

Introduction

Egomotion awareness plays a vital role in developing perception, cognition, and motor control for mobile agents through their own sensory experiences (Agrawal, Carreira, and Malik 2015). Inertial information processing, a typical egomotion awareness process operating in the human vestibular system (Cullen 2012) contributes to a wide range of daily activities. Modern micro-electro-mechanical (MEMS) inertial measurements units (IMUs) are analogously able to sense angular and linear accelerations - they are small, cheap, energy efficient and widely employed in smartphones, robots and drones. Unlike other commonly used sensor modalities, such as GPS, radio and vision, inertial measurements are completely egocentric and as such are far less environment dependent e.g. they work equally well in an unlit underground tunnel as in open spaces. Developing accurate inertial tracking is thus of key importance for robot/pedestrian navigation and for self-motion estimation (Harle 2013). In the emergency scenarios, it can help track firefighters and other first-responders to enhance the safety and efficiency of their operations without the need of any bespoke positioning infrastructure. However, the task of turning inertial measurements into pose and odometry estimates is hugely complicated by the fact that different placements (e.g. carrying a smartphone in a pocket or in the hand) and orientations lead to significantly different inertial data in the sensor frame. It is clearly infeasible to collect labelled data from every possible attachment, as this requires specialized motion capture systems e.g. VICON and a high degree of effort. In this paper, therefore, we propose a robust generative adversarial network for sequence domain transformation which is able to directly learn inertial tracking in unlabelled domains without using any paired sequences.

Prevailing inertial tracking methods, e.g. Strapdown Inertial Navigation System (SINS) (Savage 1998) and Pedestrian Dead Reckoning (PDR) (Xiao et al. 2015), are mostly based on delicate handcrafted models. These model-based approaches can obtain plausible achievements in general scenarios, but their lack of generalisation ability yields poor performance in the complex real world applications. Recent work in neural inertial tracking (Chen et al. 2018a) has demonstrated that deep neural networks are capable of extracting high level motion representations (displacement and heading angle) from raw IMU sequence data, and providing accurate trajectories. However, the data-driven method that requires substantial labeled data for training, and a model trained on a single domain-specific dataset may not generalise well to new domains (Tzeng et al. 2017). As shown in Figure 1, the uncertainties of phone placements, the corresponding motion dynamics, and the projection of gravity significantly alter the inertial measurements acquired from different domains (sensor frames) while the actual trajectories in the navigation frame are identical.

We note that it is possible to train end-to-end deep neural networks when presented with large amounts of labelled data. The question becomes, how can we generalize to an arbitrary attachment in the absence of labels or a paired/time-synchronized sequence? Although from the observation the raw inertial data for each domain is very different, and the resulting odometry trajectories are also unrelated to one another, the underlying statistical distribution of odometry pose updates, if derived from a common agent (e.g. human motion), must be similar. Our intuition is to decompose the raw inertial data into a domain-invariant semantic representation, learning to discard the domain-specific motion sequence transformation.

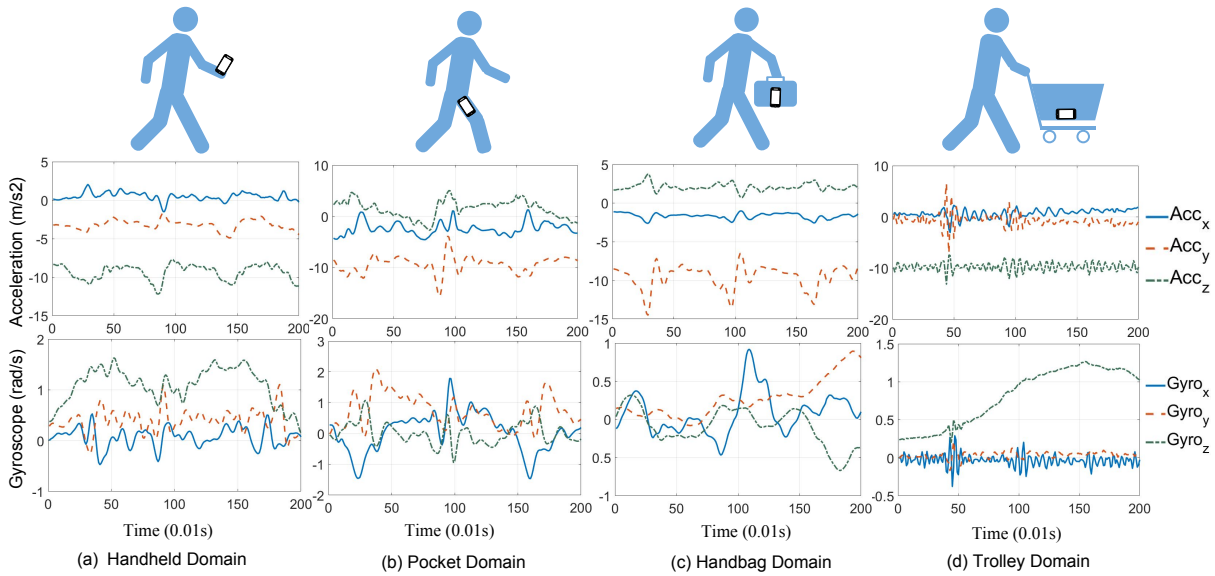


Figure 1: Phone was placed in (a) hand, (b) pocket, (c) bag and (d) trolley. Compared to firmly holding in the hand, the IMU experiences slight swings in pocket. The angular and linear accelerations in the navigation frame are projected on different axes in each sensor frame, and the gravity is mainly projected onto the y axis rather than z axis. These variations are termed motion domain shifts as the sensor frames are different yet the inertial data in the navigation frame is invariant, which impose huge challenges on transferring a learned inertial tracking system to a new domain.

To overcome the challenges of generalising inertial tracking across different motion domains, we propose the **MotionTransformer** framework with Generative Adversarial Networks (GAN) for sensory sequence domain transformation. Its key novelty is in using a shared encoder to transform raw inertial sequences into a domain-invariant hidden representation, without the use of any paired data. Different from many GAN-based sequence generation models applied in the field of natural language processing (Yu et al. 2017), where the sequences consist of discrete symbols or words (e.g. dialogue generation, poem generation and unsupervised machine translation) (Li et al. 2017), our model is focused on transferring continuous long time series sensory data. It is worth mentioning that instead of using the conditional autoregressive decoder that takes whole source sequences as input and generates variant-length sequences (generally applied in sequence-to-sequence generation models), our framework is able to take the advantage of time consistency and directly produces the outputs in target domains aligned to the inputs in source domains in every time step.

MotionTransformer dramatically reduces the effort in converting raw inertial data to an accurate trajectory, as no labelled or even paired data is required to achieve motion transformation in new domains. Through extensive, real-world experiments, we demonstrate that the framework is able to efficiently and effectively transform an arbitrary domain into an accurate inertial trajectory, benefiting from the knowledge transferred from the labelled source domain. This work addresses a challenging problem in inertial egomotion inference.

Model

Instead of directly predicting the trajectories conditioned on IMU outputs, we incorporate the neural model with a physical model for better inertial tracking inference. Here we introduce the physical model for inertial tracking and the MotionTransformer for sequence domain adaptation respectively.

Inertial Tracking Physical Model

The physical model, derived from Newtonian Mechanics, integrates the angular rates of the sensor frame $\{\mathbf{w}_i\}_{i=1}^N$ ($\mathbf{w}_i \in \mathbb{R}^3$ and N is the length of the whole sequence) measured by the three-axis gyroscope into orientation attitudes. While the linear accelerations of the sensor frame $\{\mathbf{a}_i\}_{i=1}^N (\mathbf{a}_i \in \mathbb{R}^3)$ measured by the three-axis accelerometer are transformed to the navigation frame and doubly integrated to give the position displacement, which discards the impact of the constant acceleration of gravity. This physical model is hard to implement directly on low-cost IMUs, because even a small measurement error will be exaggerated exponentially through the integration. Recent deep-learning based inertial tracking (Chen et al. 2018a) breaks the continuous integration by segmenting the sequence of inertial measurements $\{(\mathbf{a}_i, \mathbf{w}_i)\}_{i=1}^N$ into subsequences. We denote a subsequence as $\mathbf{x} = \{(\mathbf{a}_i, \mathbf{w}_i)\}_{i=1}^n$, whose length is n. By taking into subsequences as inputs, a recurrent neural network (RNN) is leveraged to periodically predict the polar vector $\mathbf{y} = (\Delta l, \Delta\psi)$, which represents the heading and location displacement:

$$(\Delta l, \Delta\psi) = \text{RNN}(\{(\mathbf{a}_i, \mathbf{w}_i)\}_{i=1}^n) \quad (1)$$

Based on the predicted $(\Delta l, \Delta\psi)$, we are able to easily construct the trajectories. However, it requires a large labelled dataset to build an end-to-end inertial tracking system, and it is infeasible to label data for every possible domain due to the motion dynamics and unpredictability of device placements. Therefore, we introduce the MotionTransformer framework in next subsection which is able to exploit the unlabelled sensory measurements in new domains and carry out accurate inertial tracking.

MotionTransformer Framework

As Figure. 2 illustrates, our framework consists of encoder, generator, decoder and predictor modules. Assume a scenario of two domains: a source domain and a target domain, where the source domain has labelled sequences $(\mathbf{x}^S, \mathbf{y}^S) \in \mathbb{D}^S$ ($\mathbf{y}^S$ is the sequence label - the polar vector of $\mathbf{x}^S$), and the target domain only has unlabelled sequences $\mathbf{x}^T \in \mathbb{D}^T$. Note that the sequences $\mathbf{x}^S$ and $\mathbf{x}^T$ are not aligned. The objectives of MotionTransformer Framework are three-fold: 1) extracting domain-invariant representations $\mathbf{z}$ shared across domains; 2) generating $\hat{\mathbf{x}}^T$ in the the target domain conditioned on $\mathbf{x}^S$; 3) predicting sequence labels $\mathbf{y}^T$ in the target domain.

Sequence Encoder To extract the domain-invariant hidden representations $\mathbf{z}$ of sensory sequences across different domains, a RNN encoder is employed together with a specific domain vector θ:

$$\mathbf{z} = f_{enc}(\mathbf{x}, \theta) \tag{2}$$

where $\mathbf{z}_i$ is aligned to $\mathbf{x}_i$ at every ith time step, and θ remains the same across all the time steps. For different domains, we apply different domain vectors θ that attempt to isolate domain-specific features, while the parameters of f_{enc} are shared across all the domains.

GAN Generator Having the domain-invariant representations $\mathbf{z}$, a RNN generator $f^T_{gen}(\mathbf{z})$ can be directly built to generate synthetic sequences $\hat{\mathbf{x}}^T$ in the target domain from $\mathbf{x}^S$. By combining it with the encoder, we derive the sequence transformation model $G_{S\to T} = f^T_{gen} \circ f_{enc}$ as:

$$\hat{\mathbf{x}}^T = G_{S\to T}(\mathbf{x}^S, \theta^S) = f^T_{gen}(f_{enc}(\mathbf{x}^S, \theta^S)) \tag{3}$$

Likewise, we construct an inverse mapping $G_{T\to S} = f^S_{gen} \circ f_{enc}$ for generating $\hat{\mathbf{x}}^S$ from $\mathbf{x}^T$:

$$\hat{\mathbf{x}}^S = G_{T\to S}(\mathbf{x}^T, \theta^T) = f^S_{gen}(f_{enc}(\mathbf{x}^T, \theta^T)) \tag{4}$$

$G_{S\to T}$ is trained against a target domain discriminator D^T (discriminators are omitted in Figure [2]) in the framework of GAN, and vice versa for $G_{T\to S}$. Unlike conventional GAN models, we decompose the generator by the domain-invariant representation $\mathbf{z}$. Intuitively, this architecture encourages the encoder f_{enc} to capture domain-invariant features that generate sensory sequences in different domains, as the encoding function f_{enc} are shared by both $G_{S\to T}$ and $G_{T\to S}$.

Reconstruction Decoder In addition to the GAN generator, we introduce a RNN decoder f_{dec} to reconstruct the sequences $\check{\mathbf{x}}$ conditioned on $\mathbf{z}$. This is aimed at reinforcing the learning of domain-invariant features when jointly learned with the GAN generator. Instead of using the conventional Denoising Autoencoders (DAE) (Vincent et al. 2010) and Variational Autoencoders (VAE) (Kingma and Welling 2014), we only introduce an additive noise to the hidden representations $\bar{\mathbf{z}} = \mathbf{z} + \epsilon$, where $\epsilon \sim N(0, I^2)$, in order to simplify the learning of the autoencoder component. Similar to the sequence encoder f_{enc}, the decoder is shared across all the domains and the domain vector θ is concatenated with inputs at every time step:

$$\check{\mathbf{x}} = f_{dec}(\bar{\mathbf{z}}, \theta) = f_{dec}(f_{enc}(\mathbf{x}, \theta) + \epsilon, \theta) \tag{5}$$

Polar Vector Predictor Since the source domain has labels (polar vectors) aligned to every sensory sequence, it is straightforward to learn a predictor for carrying out inertial tracking by supervised learning the labels $\mathbf{y}^S$. However, this is not the ultimate objective of this paper, instead we aim at transferring the knowledge learned in the labelled source domain to the unlabelled target domain. Hence, with the help of the sequence encoder f_{enc}, we construct a predictor f_{pred} also shared by both the source domain and the target domain. In this case, though there exists no paired data $(\mathbf{x}^T, \mathbf{y}^T)$ for supervised learning in the target domain, we can still predict $\mathbf{y}^T$ by:

$$\mathbf{y}^T = f_{pred}(f_{enc}(\mathbf{x}^T, \theta^T)) \tag{6}$$

Inference

This section introduces the learning method for jointly training the modules of our MotionTransformer, including GAN loss $\mathcal{L}_G$, reconstruction loss $\mathcal{L}_{AE}$, prediction loss $\mathcal{L}_{pred}$, cycle-consistency $\mathcal{L}_{cycle}$ and perceptual consistency $\mathcal{L}_{percep}$:

$$\mathcal{L}_{total} = \mathcal{L}_{GAN} + \lambda_1\mathcal{L}_{AE} + \lambda_2\mathcal{L}_{pred} + \lambda_3\mathcal{L}_{cycle} + \lambda_4\mathcal{L}_{percep} \tag{7}$$

where λ_1, λ_2, λ_3, and λ_4 are the hyper-parameters used as the trade-off for the optimization process.

GAN Loss GAN generator is one of the most important component in our framework, which is responsible of producing sensory sequences in the unlabelled target domain. Here, following the general GAN framework, we construct a discriminator D^T for the corresponding target domain and learn to discriminate the generated data $\hat{\mathbf{x}}$ from the real one $\mathbf{x}$. The GAN loss for the target domain generator can be defined as:

$$\begin{aligned}\mathcal{L}_{G^T} =& \mathbb{E}_{\mathbf{x}^T\sim p(\mathbf{x}^T)}[\log D^T(\mathbf{x}^T)] + \\ & \mathbb{E}_{\mathbf{x}^S\sim p(\mathbf{x}^S)}[\log(1 - D^T(G_{S\to T}(\mathbf{x}^S, \theta^S))]\end{aligned} \tag{8}$$

Similarly, the GAN loss for the source domain generator is:

$$\begin{aligned}\mathcal{L}_{G^S} =& \mathbb{E}_{\mathbf{x}^S\sim p(\mathbf{x}^S)}[\log D^S(\mathbf{x}^S)] + \\ & \mathbb{E}_{\mathbf{x}^T\sim p(\mathbf{x}^T)}[\log(1 - D^S(G_{T\to S}(\mathbf{x}^T, \theta^T))]\end{aligned} \tag{9}$$

Then, we combine these two losses into the final GAN loss $\mathcal{L}_{GAN} = \mathcal{L}_{G^S} + \mathcal{L}_{G^T}$.

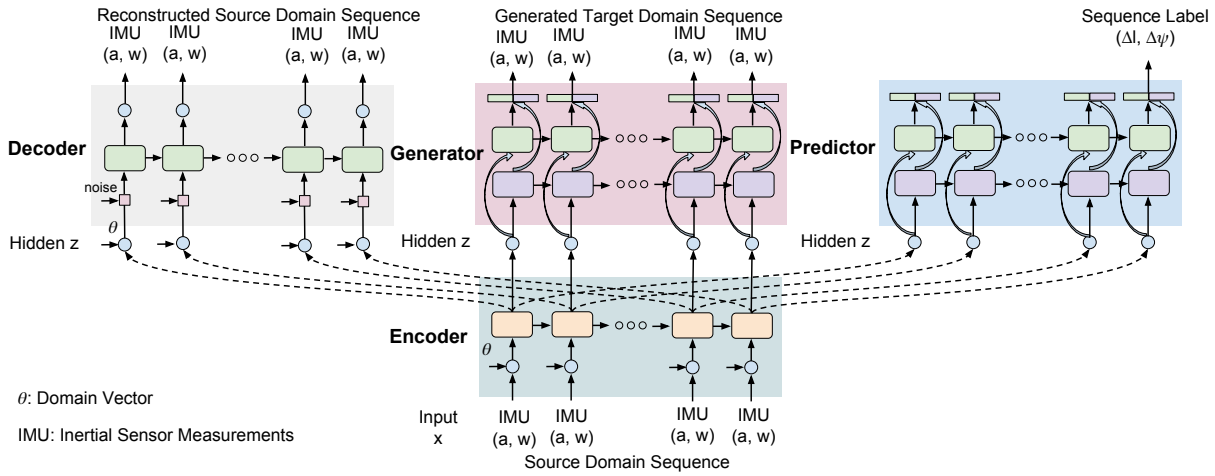


Figure 2: Architecture of Proposed MotionTransformer: including the source domain sequence **Encoder** (extracting common features across different domains), the target domain sequence **Generator** (generating sensory stream in the target domain), the sequence reconstruction **Decoder** (reconstructing the sequence for learning better representations) and the polar vector **Predictor** (producing consistent trajectory for inertial navigation). The GAN discriminators and the source domain Generator are omitted from this figure.

Reconstruction Loss Considering the inputs are the continuous real-valued data, the MSE loss is chosen to optimize the autoencoder loss for source and target domain data respectively:

$$\mathcal{L}_{AE} = \mathbb{E}_{\mathbf{x}^S \sim p(\mathbf{x}^S), \mathbf{x}^T \sim p(\mathbf{x}^T)}[\|\breve{\mathbf{x}}^S - \mathbf{x}^S\|_2 + \|\breve{\mathbf{x}}^T - \mathbf{x}^T\|_2] \tag{10}$$

Prediction Loss In addition to the original paired data $(\mathbf{x}^S, \mathbf{y}^S)$ in the source domain, we are able to make use of the generated ones $\hat{\mathbf{x}}^T = f^T_{gen}(f_{enc}(\mathbf{x}^S, \theta^S))$ produced by the GAN generator in the target domain as well, since the domain-invariant representations can be directly applied for the prediction no matter which domain the sequences are from. Hence, a joint regression loss can be constructed for learning the predictor:

$$\begin{aligned}\mathcal{L}_{pred} =& \mathbb{E}_{(\mathbf{x}^S, \mathbf{y}^S) \sim p(\mathbf{x}^S, \mathbf{y}^S)}[\|\mathbf{y}^S - f_{pred}(f_{enc}(\mathbf{x}^S, \theta^S))\|_2 + \\ & \|\mathbf{y}^S - f_{pred}(f_{enc}(\hat{\mathbf{x}}^T, \theta^T))\|_2]\end{aligned} \tag{11}$$

Although the adversarial training is unable to produce exact the same sequences as the ones generated in the target domain, the labels in the source domain encourages the sequence encoder to preserve the prominent features for prediction, so that the domain-invariant representations will be further regularised by the labels in the source domain.

Cycle Consistency Regularisation In order to improve the sensory sequence generation, we apply the cycle-consistency regularisation to ensure the sequences generated to the target domain from the source domain can be mapped back without losing too much content information. As demonstrated by (Kim et al. 2017; Zhu et al. 2017; Yi et al. 2017), this bidirectional architecture encourages the GAN to generate data in meaningful direction by punishing the optimizer with the L-1 consistency loss defined as:

$$\begin{aligned}\mathcal{L}_{cycle} =& \mathbb{E}_{x^S \sim p(x^S)} \|G_{T \to S}(G_{S \to T}(\mathbf{x}^S, \theta^S), \theta^T) - \mathbf{x}^S\|_1 + \\ & \mathbb{E}_{x^T \sim p(x^T)} \|G_{S \to T}(G_{T \to S}(\mathbf{x}^T, \theta^T), \theta^S) - \mathbf{x}^T\|_1\end{aligned} \tag{12}$$

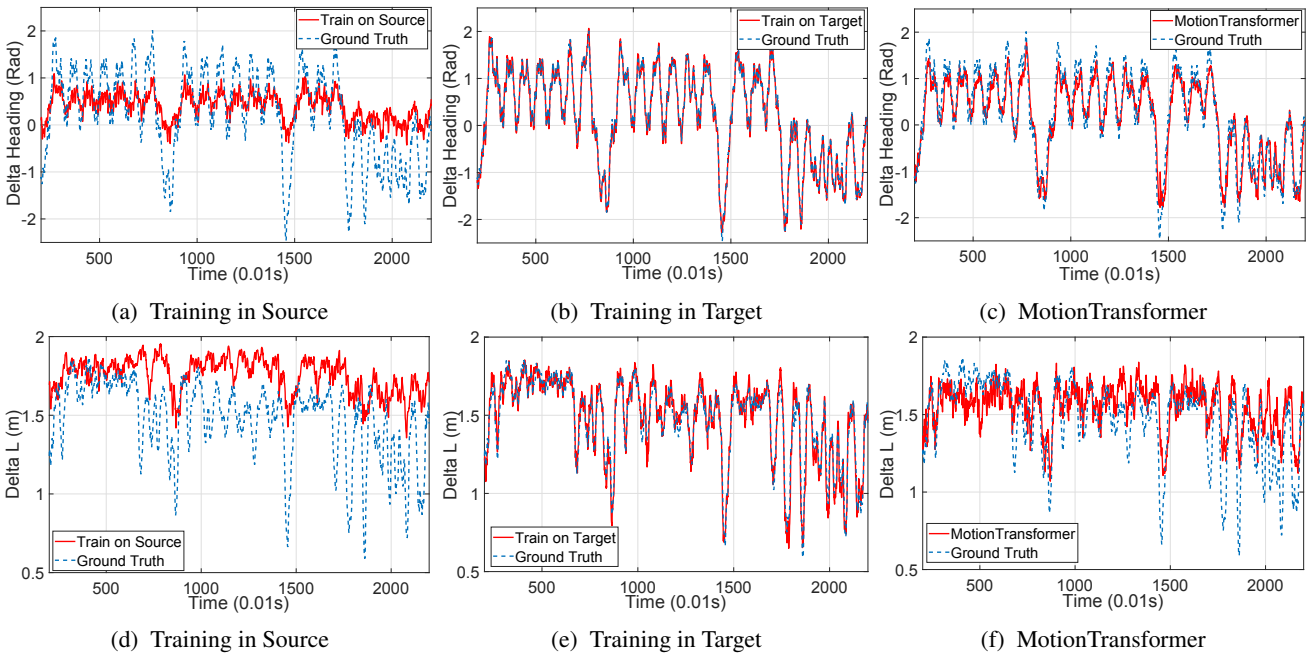
Perceptual Consistency Regularisation To further regularise the learning of domain-invariant representations, we propose the perceptual consistency regularisation. Inspired by the f constancy (Taigman, Polyak, and Wolf 2017), we employ the encoder f_{enc} as the perceptual function to enforce the semantic representation constant after being transformed into another domain by generators. For example, in source domains, the hidden representation $\mathbf{z}^S = f_{enc}(\mathbf{x}^S, \theta^S)$ extracted by the encoder conditioned on the source domain vector, will be encouraged invariant under $G_{S \to T}$ by minimizing a L-2 distance between the original hidden representation $\mathbf{z}^S$ and the hidden representation $\hat{\mathbf{z}}^S = f_{enc}(\hat{\mathbf{x}}^T, \theta^T)$ extracted from the generated synthetic target domain data $\hat{\mathbf{x}}^T = G_{S \to T}(x^S, \theta^S)$. Similar perceptual constraint can be applied for target domain.

$$\begin{aligned}&\mathcal{L}_{percep} = \\ &\mathbb{E}_{x^S \sim X^S} \|f_{enc}(x^S, \theta^S) - f_{enc}(G_{S \to T}(x^S, \theta^S), \theta^T)\|_2 \\ &+ \mathbb{E}_{x^T \sim X^T} \|f_{enc}(x^T, \theta^T) - f_{enc}(G_{T \to S}(x^T, \theta^T), \theta^S)\|_2\end{aligned} \tag{13}$$

Experiments

Inertial Tracking Dataset

A commercial-off-the-shelf smartphone, the iPhone 7Plus, is employed to collect inertial measurement data of pedestrian random walking. The smartphone was attached in four



(a) Training in Source (b) Training in Target (c) MotionTransformer

(d) Training in Source (e) Training in Target (f) MotionTransformer

Figure 3: Heading displacement estimation from training in (a) source domain, (b) target domain and (c) MotionTransformer, and location displacement estimation from training in (d) source domain, (e) target domain and (f) MotionTransformer

different poses: handheld, pocket, handbag and trolley, each of which represents a domain that has dramatically distinct motion pattern with others.

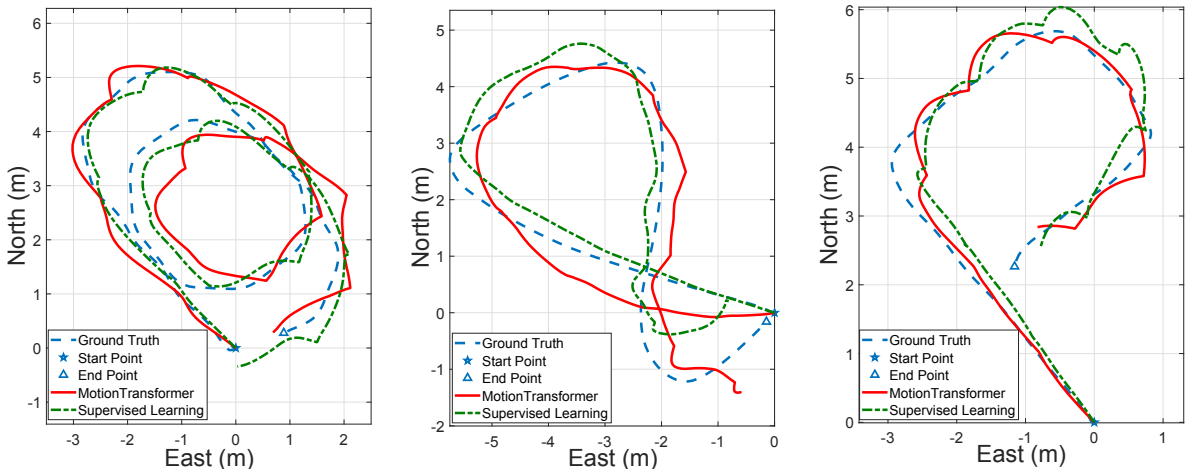
We use an optical motion capture system (Vicon) (Vicon 2017) to record the ground truth. The Vicon system is able to provide high-precision full pose reference (0.01 m for location, 0.1 degree for orientation), and tracks our participants carrying the experimental device attached with Vicon markers. The 100 Hz sensor readings are then segmented into sequences with corresponding labels, e.g. location and heading attitude displacement provided by Vicon system. These source-domain labels are used for MotionTransformer training while the target-domain labels are used for MotionTransformer evaluation only. The length of each sequence is 200 frames (2 seconds), including three linear accelerations and three angular rates per frame. In summary, the dataset[1] (Chen et al. 2018b) used in this work contains around 45K, 53 K, 36K and 29K sequences for handheld, pocket, bag, trolley domains respectively. Among them, 4K sequences were selected as validation data in each domain, and the rest was taken as training set. In our training phase, we set the hyper-parameters $\lambda_1 = 0.01$, $\lambda_2 = 100$, $\lambda_3 = 0.1$, and $\lambda_4 = 1$.

Transferring Across Motion Domains

We evaluate our model on unsupervised motion domain transfer tasks. The source domain is the inertial data collected in the handheld attachment, while the target domains are those collected in the attachments of pocket, handbag and trolley. We test the our framework with the real target data. Its generalization performance is evaluated by comparing the label prediction (polar vector) with the ground-truth data captured by Vicon system. We compare with source-only, where we use the trained source predictor to predict data directly in the target domain and with target-only where we train the target dataset with target labels (40K) to show the performance of fully supervised learning. Figure 3 presents the predicted location and heading displacement in pocket domain for the three different techniques. It can be seen that source-only is unable to follow either delta heading or delta location accurately, whereas MotionTransformer achieves a level of performance close to the fully supervised target-only, especially for delta heading.

Table. 1 presents the *quantitative analysis* with a metric of mean square error of the label prediction against the ground truth. Compared with using a model trained on source data only, our proposed unsupervised sequence domain adaptation technique helps to dramatically decrease the validation loss (almost 6 times, 9 times, and 3 times in pocket, bag and trolley domains). Two popular baselines are compared with MotionTransformer: i) adversarial discriminative domain adaptation (ADDA) (Tzeng et al. 2017) and ii) cycle-consistent adversarial domain adaptation (CyCADA) (Hoffman et al. 2017). ADDA is a discriminative model, forcing the feature fusion of two domains by distinguishing the features after the encoder. CyCADA is a generative model, using standard Cycle-GAN framework to generate synthetic target domain data and fine-tune the predictor through synthetic data. Because they both aim to process images rather than continuous sequential data, we replace their convolutional generator and discriminator with the same LSTM layers as described in our frameworks. Moreover, we cut down the reconstruction loss and percep-

[1]Dataset can be found at http://deepio.cs.ox.ac.uk



(a) Pocket (b) Trolley (c) Bag

Figure 4: Inertial tracking trajectories of (a) Pocket (b) Trolley (c) Handbag, comparing our proposed unsupervised MotionTransformer with Ground Truth and Supervised Learning.

Table 1: Unsupervised Transfer Across Motion Domains

Model	Hand → Pocket	Hand → Bag	Hand → Trolley
Training on Source, Testing on Target	0.718	1.129	0.273
ADDA with LSTM	0.471	0.631	0.216
CyCADA with LSTM	0.237	0.455	0.182
MotionTransformer w/o Reconstruction	0.313	0.335	0.113
MotionTransformer w/o Perceptual Loss	0.315	0.202	0.140
MotionTransformer	**0.119**	**0.123**	**0.098**
Semi-Supervised MotionTransformer (1K)	0.113	0.075	0.047
Train on Target, Test on Target (40K)	0.045	0.010	0.006

tual loss in our framework respectively to show their impact. As shown in Table. 1, our MotionTransformer still achieves competitive performance compared to these baselines.

Lastly, we also evaluate our framework on a semi-supervised domain transfer task, with 1K labelled target domain sequences to train the predictor. As shown in Table. 1, the sparse labels in target domains help decrease the prediction error, especially in the bag and trolley domains.

Inertial Tracking in Unlabelled Domains

We argue that the predicted label from our domain transformation framework is capable of solving a downstream task - inertial odometry tracking. In an inertial tracking task, the precision of the predicted label determines the localization accuracy, as the current location (x_n, y_n) is calculated by using an initial location (x_0, y_0) and heading, and chaining the results of previous windows via Equation 14. This dead reckoning (DR) technique, also called path integration, can be widely found in animal navigation (McNaughton et al. 2006), which enables animals to use inertial cues (e.g. steps and turns) to track themselves in the absence of vision. The errors in path integration will accumulate and cause unavoidable drifts in trajectory estimation, which imposes a requirement for accurate motion domain transformation. Without domain adaptation, if the model trained on source domain is directly applied to data from target domains, it will not produce any trajectory representing the self-motion. When using only inertial information, traditional model-based navigation algorithms perform poorly in unlabelled scenarios, as SINS will collapse due to the high measurements noises, and PDR will be influenced by incorrect step detection or device orientation.

$$\begin{cases} x_n = x_0 + \Delta l cos(\psi_0 + \Delta\psi) \\ y_n = y_0 + \Delta l sin(\psi_0 + \Delta\psi) \end{cases} \tag{14}$$

We show that the inertial tracking trajectory can be recovered from the labels predicted by our domain adaptation framework in *unlabelled* domains. The participant walked with the device placed in the pocket, the handbag and on the trolley. The inertial data during test walking trajectory was not included in training dataset, and collected in different days. Figure 4 illustrates that our proposed model succeeds in generating physically meaningful trajectories, close to the ground truth captured by Vicon system. It proves that exploiting the raw sensory stream and transforming to a common latent distribution can extract meaningful semantic features that help solve downstream tasks.

Figure 5: Visualization of extracted representations in the source and target domains. It can be seen that the MotionTransformer leads to a more consistent latent representation compared with the disjoint representations of the normal encoder and the ADDA encoder.

Interpreting the Sequence Encoder

The role of the sequence encoder is evaluated by the t-SNE projection to show its ability to map the raw data from two domains to an identical semantic space. We compare it with two other baselines: a domain-specific encoder, which is only employed in the source domain (it is not shared across domains); an ADDA encoder, which is learned jointly with the predictor by adversarial training to force the fusion of representations extracted by the encoder. The t-SNE projection is shown in Figure 5, and all of the models apply the same parameters (Perplexity=10, step=5000). As can be seen, the data points of domain-specific encoder are distinctly separated into two folds. ADDA attempts to fuse the points from two domains but it turns out points are still clearly separated. By contrast, the encoder of our MotionTransformer is able to better scatter the points dispersively in the semantic space, which removes the domain shifts and benefit target label prediction.

Related Work

Domain Adaptation Our work is most related to domain adaptation techniques, which aim to align the learned representation across source and target domains by minimizing maximum mean discrepancy loss (Long et al. 2015) or adversarial loss (Ganin et al. 2016; Tzeng et al. 2017). Recent adversarial approaches have been achieved the state of art results in multiple tasks, for example, sleep stages prediction (Zhao et al. 2017), healthcare data prediction (Purushotham et al. 2017) and image-to-image translation (Liu, Breuel, and Kautz 2017). Prior art can be categorized into two main groups: the discriminative adversarial models seek to align the embedding representation between target and source domain to encourage domain confusion (Ganin et al. 2016; Tzeng et al. 2015; 2017); the generative adversarial models aim to employ generated data for training the prediction networks and meanwhile fooling the discriminator (Shrivastava et al. 2017; Hoffman et al. 2017; Liu, Breuel, and Kautz 2017). Here, we utilize the generative adversarial networks (GAN) to generate sensory sequence data from invariant features extracted by an identical encoder.

Inertial Navigation Systems Early inertial navigation systems were developed as the core components in control and navigation systems for missiles, submarines, and spacecraft, relying on expensive, heavy and high-precision inertial measurement units (Savage 1998). The traditional strapdown inertial navigation algorithms are hard to realize on low-cost MEMS inertial sensors, because the high measurement noises cause exponential error propagation via open integration, and the inertial output collapses within seconds (Harle 2013). To mitigate the unbounded error drift, one solution is to combine cameras with inertial sensors as realized in visual inertial odometry (Li and Mourikis 2013). Another solution is to detect steps and update the trajectory with estimated step length and heading through pedestrian dead reckoning (Xiao et al. 2015). These model based approaches exploit the context information to reduce the inertial systems drift, for example, via zero-velocity update (Nilsson et al. 2012; Chen et al. 2016), floor map (Xiao et al. 2014), electronic magnetic field (Lu et al. 2018), but their assumptions are too strong. As a consequence their performance is variable in complex real-world conditions: visual-inertial odometry assumes cameras have feature-rich, well illuminated scenes without occlusion, and PDR assumes the user's personal walking model and the phone placement are prior knowledge (Brajdic and Harle 2013). Recent deep learning based inertial tracking (Chen et al. 2018a) can learn direct location transforms from raw inertial data, and construct continuous accurate trajectories for indoor users, but still suffers from serious domain shifts and generalization problems. Our work aims to unsupervised learn the inertial tracking in unlabeled new domains, effectively increasing its generalization ability and flexibility in real usages.

Conclusion and Discussion

Motion transformation between different domains is a challenging task, which typically requires the use of labeled data for training. In the presented framework, by transforming target domains to a consistent, invariant representation, a physically meaningful trajectory can be well reconstructed. Intuitively, our technique is learning how to transform data from an arbitrary sensor domain θ to a common latent representation. Analogously, this is equivalent to learning how to translate any sensor frame to the navigation frame, without any labels in the target domain. Although MotionTransformer has been shown to work on IMU data, the broad framework is likely to be suitable for any continuous, sequential domain

transformation task where there is an underlying physical model.

Acknowledgements

We thank all the reviewers and ACs. This work is funded by the National Institute of Standards and Technology (NIST) Grant No. 70NANB17H185. We also hope to thank Prof. Xiaoping Hu, Prof. Xiaofeng He, and Prof. Lilian Zhang at National University of Defense Technology, China for their useful assistance and valuable discussion, who are supported by the National Natural Science Foundation of China (Grants Nos. 61773394, 61573371, 61503403).

References

Agrawal, P.; Carreira, J.; and Malik, J. 2015. Learning to see by moving. In *ICCV*, volume 11-18, 37–45.

Brajdic, A., and Harle, R. 2013. Walk detection and step counting on unconstrained smartphones. In *Ubicomp*.

Chen, C.; Chen, Z.; Pan, X.; and Hu, X. 2016. Assessment of zero-velocity detectors for pedestrian navigation system using mimu. In *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, 128–132.

Chen, C.; Lu, X.; Markham, A.; and Trigoni, N. 2018a. IONet: Learning to Cure the Curse of Drift in Inertial Odometry. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI)*.

Chen, C.; Zhao, P.; Lu, C. X.; Wang, W.; Markham, A.; and Trigoni, N. 2018b. OxIOD: The Dataset for Deep Inertial Odometry. *arXiv* 1809.07491.

Cullen, K. E. 2012. The vestibular system: Multimodal integration and encoding of self-motion for motor control. *Trends in Neurosciences* 35(3):185–196.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research* 17:1–35.

Harle, R. 2013. A Survey of Indoor Inertial Positioning Systems for Pedestrians. *IEEE Communications Surveys and Tutorials* 15(3):1281–1293.

Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.-Y.; Isola, P.; Saenko, K.; Efros, A. A.; and Darrell, T. 2017. CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In *ICML*, 1–12.

Kim, T.; Cha, M.; Kim, H.; Lee, J. K.; and Kim, J. 2017. Learning to Discover Cross-Domain Relations with Generative Adversarial Networks. In *ICML*.

Kingma, D. P., and Welling, M. 2014. Auto-Encoding Variational Bayes. In *ICLR*, 1–14.

Li, M., and Mourikis, A. I. 2013. High-precision, consistent EKF-based visual-inertial odometry. *The International Journal of Robotics Research* 32(6):690–711.

Li, J.; Monroe, W.; Shi, T.; Jean, S.; Ritter, A.; and Jurafsky, D. 2017. Adversarial Learning for Neural Dialogue Generation. In *EMNLP*.

Liu, M.-Y.; Breuel, T.; and Kautz, J. 2017. Unsupervised Image-to-Image Translation Networks. In *NIPS*.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. I. 2015. Learning Transferable Features with Deep Adaptation Networks. In *ICML*, volume 37, 1–20.

Lu, C. X.; Li, Y.; Zhao, P.; Chen, C.; Xie, L.; Wen, H.; Tan, R.; and Trigoni, N. 2018. Simultaneous localization and mapping with power network electromagnetic field. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, MobiCom '18, 607–622. New York, NY, USA: ACM.

McNaughton, B. L.; Battaglia, F. P.; Jensen, O.; Moser, E. I.; and Moser, M. B. 2006. Path integration and the neural basis of the 'cognitive map'. *Nature Reviews Neuroscience* 7(8):663–678.

Nilsson, J. O.; Skog, I.; Händel, P.; and Hari, K. V. S. 2012. Foot-mounted INS for everybody - An open-source embedded implementation. In *IEEE PLANS, Position Location and Navigation Symposium*, 140–145.

Purushotham, S.; Carvalho, W.; Nilanon, T.; and Liu, Y. 2017. Variational Recurrent Adversarial Deep Domain Adaptation. In *ICLR*, 1–11.

Savage, P. G. 1998. Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms. *Journal of Guidance, Control, and Dynamics* 21(1):19–28.

Shrivastava, A.; Pfister, T.; Tuzel, O.; Susskind, J.; Wang, W.; and Webb, R. 2017. Learning from Simulated and Unsupervised Images through Adversarial Training. In *CVPR*.

Taigman, Y.; Polyak, A.; and Wolf, L. 2017. Unsupervised Cross-Domain Image Generation. In *ICLR*, 1–15.

Tzeng, E.; Hoffman, J.; Darrell, T.; Saenko, K.; and Lowell, U. 2015. Simultaneous Deep Transfer Across Domains and Tasks. In *ICCV*.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial Discriminative Domain Adaptation. In *CVPR*.

Vicon. 2017. ViconMotion Capture Systems: Viconn.

Vincent, P.; Larochelle, H.; Lajoie, I.; Bengio, Y.; and Manzagol, P.-A. 2010. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion Pierre-Antoine Manzagol. *Journal of Machine Learning Research* 11:3371–3408.

Xiao, Z.; Wen, H.; Markham, A.; and Trigoni, N. 2014. Lightweight map matching for indoor localization using conditional random fields. In *International Conference on Information Processing in Sensor Networks (IPSN)*, 131–142.

Xiao, Z.; Wen, H.; Markham, A.; and Trigoni, N. 2015. Robust indoor positioning with lifelong learning. *IEEE Journal on Selected Areas in Communications* 33(11):2287–2301.

Yi, Z.; Zhang, H.; Tan, P.; and Gong, M. 2017. DualGAN: Unsupervised Dual Learning for Image-to-Image Translation. In *ICCV*, 2868–2876.

Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *AAAI*, 2852–2858.

Zhao, M.; Yue, S.; Katabi, D.; Jaakkola, T. S.; and Bianchi, M. T. 2017. Learning Sleep Stages from Radio Signals: A Conditional Adversarial Architecture. *ICML* 70:4100–4109.

Zhu, J.-y.; Park, T.; Efros, A. A.; Ai, B.; and Berkeley, U. C. 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *ICCV*.