Wormhole Learning

Alessandro Zanardi Julian Zilly Andreas Aumiller Andrea Censi Emilio Frazzoli

***Abstract*— Typically, to enlarge the operating domain of an object detector, more labeled training data is required. We describe a method called *wormhole learning*, which allows to extend the operating domain without additional data, but only with *temporary access* to an auxiliary sensor with certain invariance properties.**

We describe the instantiation of this principle with a regular visible-light RGB camera as the main sensor, and an infrared sensor as the temporary sensor. We start with a pre-trained RGB detector; then we train the infrared detector based on the RGB-inferred labels; finally we re-train the RGB detector based on the infrared-inferred labels. After these two transfer-learning steps, the RGB detector has enlarged its operating domain by inheriting part of the invariance to illumination of the infrared sensor; in particular, the RGB detector is now able to see much better at night.

We analyze the wormhole learning phenomenon by bounding the possible gain in accuracy using mutual information properties of the two sensors and considered operating domain.

I. Introduction

A distinctive feature of humans is the ability to transfer knowledge across different sensory inputs. Famously Beethoven and other famous composers continued to write music while being deaf, suggesting that they were able to transfer some previously acquired auditory skills to the visual domain [1]. In pioneering experiments summarized in [2], Erismann and Kohler devised studies in which participants wore special mirror goggles which distort the visual field of the wearer, e.g. by flipping their visual field upside down. They discovered that humans are remarkably adept at adjusting to such distortions by relying on their other senses to the point that after a few days their perception returns to “normal" even when wearing the upside-down goggles.

In this work, we likewise focus on the ability to employ *transfer learning* across heterogeneous sensors by exploiting their inherent symmetry characteristics. Our main contribution is the conception and execution of *wormhole learning* visualized in Fig. 1, which can be regarded as a type of semi-supervised learning that leverages sensory and algorithmic symmetries. Most notably we are able to augment the original operating domain of an algorithm by retraining it based on information obtained in a different sensor space, as illustrated in Fig. 2. We analyze when and why wormhole learning is possible, with implications concerning in which domains transfer learning across sensors is possible.

Experimentally, we demonstrate that, from a pretrained object detector for regular RGB-images, it is possible to train an object detector on long wave infrared (IR) data using a joint-sensor setup (first transfer step in Fig. 1). Exploiting the illumination invariance of the IR camera, the IR object detector is then used to reliably label data in previously challenging lighting conditions at night (invariance step in Fig. 1). This is then used to retrain the RGB object detector with both *labeled* night and day data (second transfer step in Fig. 1), effectively augmenting its original operating envelope.

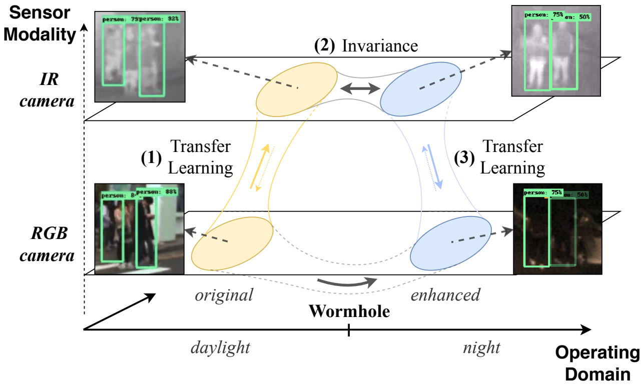
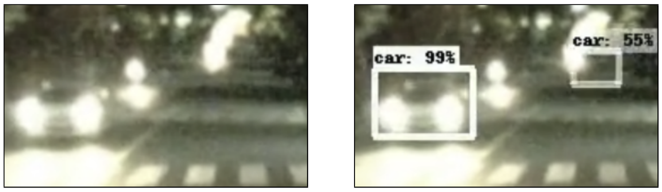


Fig. 1. A *wormhole* can be created in the operating domain of a sensor leveraging the inherent invariance of another auxiliary sensor—in this case invariance to illumination of an IR camera. Starting with an object detector trained only with daytime data we enlarge its capabilities to also include night time thanks to the *temporary* addition of an IR camera. The night-day wormhole is created from three steps: 1) Transfer learning from RGB camera to IR at daytime. 2) Exploiting the invariance of IR camera with respect to time of the day. 3) Transfer learning back from IR to RGB at night.



(a) Before *wormhole learning* (b) After *wormhole learning*

Fig. 2. Despite starting only with daytime data, and no additional labels, the *wormhole*-enhanced RGB detector has improved performance at night. For example, we observe that the detector has learned to recognize cars by their blinding lights (not present at all at daytime).

Thus *wormhole learning* is a novel method of bootstrapping the training of new sensors and most importantly generating "labeled" data in a new domain "without cost", serving as a stepping stone to more robustness and performance. We will show that the method is most useful in setups where an agent has access to heterogenous sensors that have some complementary invariant properties, such as the self-driving domain.

The authors are with the Institute for Dynamic Systems and Control (IDSC), ETH Zürich, Switzerland. Please address correspondence to {azanardi,jzilly}@ethz.ch.

II. Related work

Due to the various elements involved in *wormhole learning*, the present work touches upon three topics: multi-modal perception, transfer learning and semi-supervised learning.

1) Multi-modal perception: Oftentimes, multi-sensor setups are used to increase robustness or performance via *sensor fusion* [3]. In recent works, diverse sensor readings are oftentimes joined together using Neural Networks (NNs) such as performed for robotic multi-sensor fusion [4]. Moreover, the range of such applications spans from improved detection and tracking [5] to better prediction for successful grasping [6].

While complementarity and fusion are necessary for robustness, many works have focused on overlap and correlation between different sensing modalities. In [7], a retrieval algorithm to find correspondences between text and images is presented. Also generative adversarial networks (GANs) [8] have been recently used to learn cross-domain relationships [9] showing the ability to transfer style between different objects types, to synthesize images based on text [10] and to learn joint distributions over multi-domain images [11].

2) Transfer learning: Transfer learning is commonly interpreted as the ability to adapt learning across different distributions of data [12]. A large body of research has developed around applications of transfer learning for multiple tasks [13], [14] which include, but are not limited to, learning to play multiple Atari games [15] using the same parameters, machine translation trained on diverse language pairs which can be tested and generalized to previously unseen language pairs [16], and performing diverse robotic tasks using modular neural networks [17]. Transferring one domain to another, recently CycleGANs [18] were proposed as a way to, for example, transfer a summertime picture to winter [19] while another approach [20] was used to restore image quality in underexposed images. All these methods aim to exploit insights learned from the training distribution to augment their capabilities in a different target domain.

3) Semi- and self-supervised learning: Semi- and self-supervised learning is frequently employed to bypass the need for data labeling due to cost or difficulty in the labeling process. A range of examples include semi-supervised object recognition [21], robust tracking [22], and many more [23]. Self-supervised learning has been similarly effective, for instance in road detection [24], robotic terrain traversal cost prediction [25] and robot grasping [26].

In comparison, *wormhole learning* is a semi-supervised learning technique which hinges upon *temporary* access to an additional sensor which has complementary invariance properties compared to the first sensor. We will clarify when and how *wormhole learning* is possible, possibly providing valuable lessons for semi-supervised, self-supervised and transfer learning.

III. Problem formalization

The general problem we are addressing in this work is to enlarge the domain of a learned mapping from data to a task embedding to a new operating domain. In our particular case, we aim to show how this is possible by employing a temporary auxiliary sensor. The ensuing mathematical formalization takes inspiration from [27]–[29].

a) Scene to data: We characterize a sensor ζ as a mapping $z(\cdot)$ from a scene $\mathrm{x} \in \mathrm{X}$ to its data representation $\mathrm{z} \in \mathrm{Z}$. Note that X will be used interchangeably as random variable (RV), but also, to simply indicate a set of scenes, the context will make it clear.

b) Task: From a given sample of data z, "accomplishing a task" [30] refers to the ability to infer the corresponding label $\mathrm{y} \in \mathrm{Y}$. While in other works [27]–[29] the Markov chain $\mathrm{x} \to \mathrm{z} \to \mathrm{y}$ reads *data* $\to$ *representation* $\to$ *task*, in our multi-sensor setting it refers to *scene* $\to$ *data* $\to$ *task*.

c) Nuisance: A *nuisance* ν for a task is a RV which affects the scene but is independent of the task, i.e. $I(\nu; \mathrm{Y}) = 0$, where $I(\cdot;\cdot)$ denotes the mutual information. We consider the task of object detection which encompasses recognition and localization of predetermined classes on the image plane. Note that common nuisances such as translation and rotation are not actual disturbances in our case due to the required localization. In this work though, we take into account the changes in illumination that will be simplified to daytime vs. night (D/N can be seen as a RV affecting the data).

To facilitate further reading we summarize the necessary elements to describe *wormhole learning* in Tab. I.

TABLE I
Summary of notation adopted in wormhole learning

Symbol	Meaning
$z(\cdot) : \mathrm{X} \to \mathrm{Z}$	Sensor projection of the scene to data.
$\overline{\mathrm{Y}}$	True label distribution independent of the sensor corresponding only to X.
$\mathrm{Z}^{\mathrm{D/N}}_{\mathrm{RGB/IR}}$	Data distribution given day/night from RGB/IR camera.
$\mathrm{z} \mapsto f_\theta(\mathrm{z})$	Detector, e.g. neural network, trained to learn $p(\mathrm{Y}\vert\mathrm{Z})$.
θ, ψ	Detector parameter distributions trained on $\mathrm{Z}^{D/N}_{\mathrm{RGB/IR}}$ and associated labels.
$\mathrm{Y}^{\mathrm{D/N}}_{\mathrm{RGB/IR}}$	Task/label RV associated to the learned distribution $f_{\theta^{\mathrm{D/N}}_{\mathrm{RGB/IR}}} := q(\mathrm{Y}\vert\mathrm{Z}, \theta^{\mathrm{D/N}}_{\mathrm{RGB/IR}})$.

A. Segmentation of the scene space

The transfer learning steps in *wormhole learning* force us to closely consider the overlaps of sensor representations and their correlation with the task. We first define the concept of operating domain mathematically, then dive into a more detailed description of Fig. 3.

Definition 1 (Potential operating domain). *Let ζ be a fixed sensor and* Y *a fixed task. We denote by* $\mathrm{x} \in \mathrm{X}$ *a subset of the scene space. Hence,* $\mathrm{z} = z(\mathrm{x}) \;\; \forall \mathrm{x} \in (\mathrm{X})$. *We define the* potential operating domain *of ζ relative to* $\overline{\mathrm{Y}}$ *as*

$$\mathsf{OD}_\delta(\zeta, \mathrm{Y}) = \left\{ \mathrm{x} \in \mathrm{X} \; s.t. \; I(z(\mathrm{x}); \overline{\mathrm{Y}}) > \delta \right\},$$

where $\delta > 0$ is a desirable lower bound to the mutual information between data and the task.

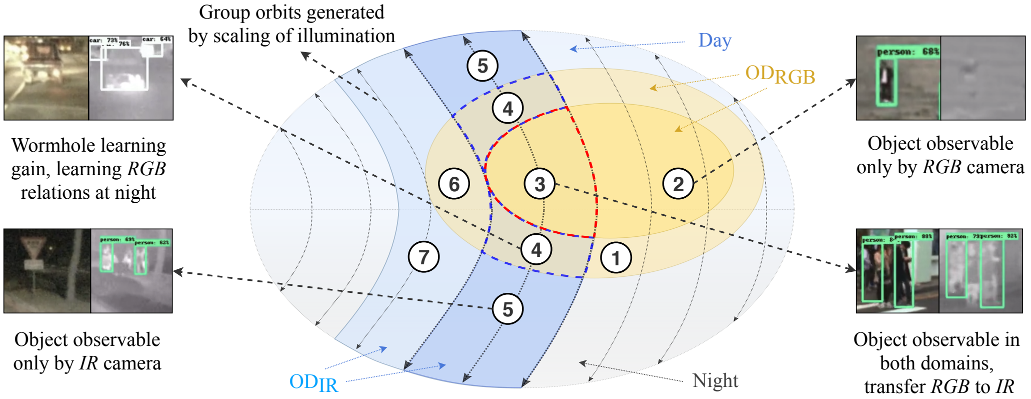


Fig. 3. We segment the space of all possible scenes into the operating domain of an RGB ($\mathsf{OD}_{\mathrm{RGB}}$) and an IR ($\mathsf{OD}_{\mathrm{IR}}$) camera. Relative to the task of object detection this reads *"observable"* objects in the images (Cf. Definition 1). We consider the dark yellow region (i.e. $3 \cup 2$) to be the subset where we sampled the initial daytime-only data. Region 3, contoured with dashed red, depicts the transfer learning region where we aim to train the IR detector. Thanks to its approximate invariance to the illumination acting on the scene space, we also gain, *without cost*, the scenes in 4 and 5 including the entire equivalence class. The last learning step of the wormhole (IR to RGB) allows us to actually expand the initial operating envelope by including region 4. For completeness, 1 is equivalent to 2 with difficult lighting condition for the RGB domain, where the object exists but its representation has not been learned by the network. Region 6 in turn can be interpreted as objects that are not of interest for our detector (e.g. a horse), while 7 depicts the region corresponding to region 1 for $\mathsf{OD}_{\mathrm{IR}}$.

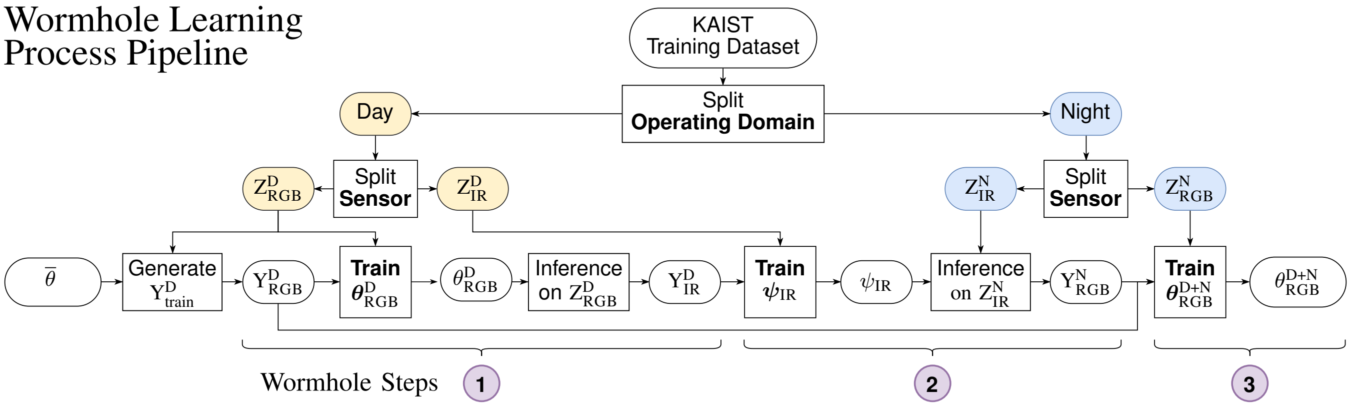


Fig. 4. The process can be generally divided into three sub-tasks: 1) Training of the network using day-time only annotations and transferring generated labels to $\mathsf{OD}_{\mathrm{IR}}$. 2) Training of the IR network using labels generated by the object detector trained in last step and exploiting invariance of the IR sensor to generate labels at night, which are transferred back to $\mathsf{OD}_{\mathrm{RGB}}$. 3) Wormhole loop is closed by retraining the original network with day-time and night-time data from the IR network.

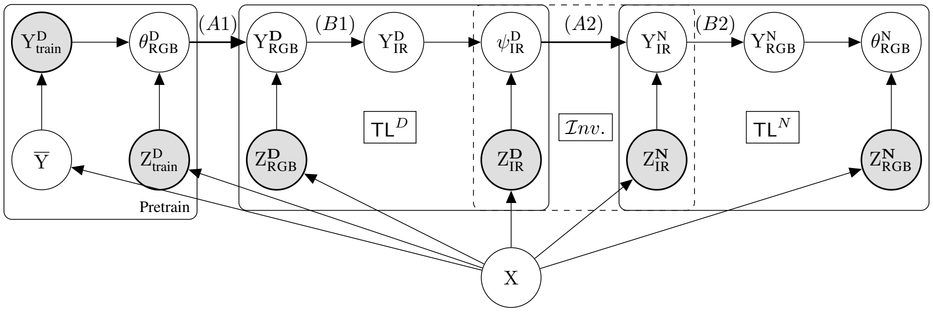


Fig. 5. Bayesian diagram of wormhole learning process. Gray denotes observable distributions. White circles refer to unobservable latent distributions. The processes (A) denote the transfer of the data distribution due to approximate invariance of the change in input distribution. The processes (B) represent the transfer of labels due to joint measurements of the same scene with distinct sensors. Exponents D and N represent whether data is sampled during daytime or at night, respectively. Note that $\theta^{\mathrm{D}}_{\mathrm{RGB}}$ has been pretrained on another dataset denoted by subscript "train".

B. Sensor properties

Nuisances acting on a scene can either have the structure of a group, and they are invertible, or can be non-invertible (e.g. occlusion). For simplicity, assume the illumination nuisance affecting the operating domain to have the form of a group. For the auxiliary IR camera, temperature is a nuisance which however does not generally affect regular cameras.

The operating domain is then visualized in Fig. 3 as a cross section of the scene space parallel to the group orbits generated by the illumination nuisance. The blue and yellow regions in Fig. 3 are an idealized depiction of these areas for IR and RGB camera relative to the object detection task. They represent an operating envelope, i.e. a subset of all possible scenes, for which the sensor generates data with *"enough"* information to solve the task.

While it is useful to illustrate the invariance as orbits of a group action, we do not need this condition, as later we define the invariance as the mutual information among the data conditioned on specific values of a nuisance variable. In the RGB/IR setup, the invariance holds exactly only locally. For example, the scaling of illumination is highly dependent on the time of day, which induces second order effects also in the infrared domain (e.g. temperature of the ambient is correlated to the time of the day).

IV. RGB/IR PIPELINE

We empirically validate the concept of *wormhole learning* on the KAIST multi-spectral dataset [31]. A synchronous stream of RGB and IR images is provided along a split between daytime (D) and night (N) data.

In order to generate the ground-truth labels for the initial daytime training and testing set, we adopted a pretrained Faster-RCNN network[1] using the NASnet architecture [32].

For reliable object detection at night, data annotations were obtained by hand-labeling a subset of the night test set. We sampled 1 out of every 7 images totaling 2281 test samples at night (Cf. Tab. IV-A for details on objects categories). Note that not a single label was used for night training, since they are automatically generated through semi-supervised transfer learning from and to the IR camera.

A. Networks

For each *wormhole learning step* in Fig. 4, we start training from the same checkpoint of a Single Shot Detector network (SSD) pretrained on COCO [33] incorporating the Inception V2 module [34]. All training sessions were carried out with a batch size of 32, RMSprop optimizer with initial learning rate of $6 * 10^{-3}$, momentum of $\beta = 0.9$, and an exponential decay factor of 0.95 after 60k steps. Additionally we applied standard data augmentation techniques [35] to increase robustness and limit overfitting; these include: random horizontal flips, random crops and random scaling in brightness and contrast. These lasts techniques have to be considered small and local with respect to the general changes in illumination.

Finally, all the evaluations have been carried out following the PASCAL VOC best practice guidelines [36]. This implies a standard 0.5 IoU score for a positive match and counting multiple detections of the same object as false positives. For more comprehensive details, code and model files for this paper will be made available via Github.

TABLE II

NUMBER OF OBJECTS IN THE DATASETS

Dataset	Car	Person	Bus	Truck	Moto.	Bicycle
Train Day	132190	42302	4430	2212	6559	26721
Train Day[a]	116907	37465	4279	1568	3952	17585
Train Night[b]	27782	9213	204	11	0	24
Test Day	150960	54219	2733	2067	493	174
Test Night[c]	4400	2254	230	1085	31	7

[a] generated by RGB detector
[b] generated by IR detector
[c] hand-labeled with labelImg [37].

B. Step 1: Training of a daytime only RGB object detector

As displayed in Fig. 4, an initial network $\theta^{\mathrm{D}}_{\mathrm{RGB}}$ is trained using ground-truth data $\mathrm{Y}^{\mathrm{D}}_{\mathrm{train}}$ for day-time only RGB images. We then deploy $\theta^{\mathrm{D}}_{\mathrm{RGB}}$ to generate inferred labels $\mathrm{Y}^{\mathrm{D}}_{\mathrm{RGB}}$ that are transferred to $\mathrm{Y}^{\mathrm{D}}_{\mathrm{IR}}$: we now have labels to train an IR detector.

C. Step 2: Training of an IR object detector

An auxiliary network ψ^{D}_{IR} is trained by feeding $\mathrm{Y}^{\mathrm{D}}_{\mathrm{IR}}$ and $\mathrm{Z}^{\mathrm{D}}_{\mathrm{IR}}$, which, upon completion, is able to detect objects in the IR domain. In a further step, we deploy the trained auxiliary network ψ^{D}_{IR} on IR night data $\mathrm{Z}^{\mathrm{N}}_{\mathrm{IR}}$ to generate $\mathrm{Y}^{\mathrm{N}}_{\mathrm{IR}}$ which are transferred to $\mathrm{Y}^{\mathrm{N}}_{\mathrm{RGB}}$.

D. Step 3: Training of a day & night RGB object detector

In this last step, we close the wormhole loop. Yet another network $\theta^{D+N}_{\mathrm{RGB}}$ is trained using joint label data $\mathrm{Y}^{\mathrm{D}}_{\mathrm{RGB}}, \mathrm{Y}^{\mathrm{N}}_{\mathrm{RGB}}$ and input data $\mathrm{Z}^{\mathrm{D}}_{\mathrm{RGB}}, \mathrm{Z}^{\mathrm{N}}_{\mathrm{RGB}}$ which yields $\theta^{D+N}_{\mathrm{RGB}}$.

E. Results

In the following we consider the object detection results detailed in Tab. III and IV.

The first two rows of Tab. III show that the initial RGB detector behaves well at day-time, but loses substantially in performance at night as highlighted by the mean average precision (mAP). After the first transfer learning, we observe in the 3rd row, that the IR detector underperforms at daytime for *bicycles, motorcycles, trucks*, and *persons*. We ascribe this to a low "observability" of these categories at daytime in the IR domain (i.e. they belong to region 2 in Fig. 3). Surprisingly, traveling along the IR invariance improves the performance as apparent from the 4th row.

Empirically we observed that the time of the day induces second order effects in the IR domain (e.g. environment temperature) and cannot be regarded as an illumination change decoupled from the rest of the scene. The result is that it enhances the contrast at night leading to a better discriminative

[1] Faster-RCNN is available as part of the Tensorflow model zoo on Github.

TABLE III

DETECTION PERFORMANCE COMPARISON FOR EACH CLASS. IMPROVED VALUES ARE MARKED IN BOLD

Network	Testset	Car	Person	Bus	Truck	Motorcycle	Bicycle	mAP
1 RGB	Day	75.86	42.90	66.89	18.77	37.41	20.53	43.74
(Day Only)	Night	44.48	21.36	13.66	0.87	0.03	0.02	13.40
2 IR[a]	Day	57.85	17.69	52.06	5.48	0.33	1.20	22.44
	Night	67.18	44.86	43.34	18.66	13.18	0.00	31.20
3 RGB[b]	Day	75.81	38.95	63.64	15.80	**40.55**	14.53	41.55
(Day+Night)	Night	**65.77**	**25.24**	**30.53**	0.00	0.00	0.00	**20.26**

[a] Network trained on labeled data generated with a retrieval threshold of 0.35

[b] Network trained on labeled data generated with a retrieval threshold of 0.50

TABLE IV

CONFIDENCE THRESHOLD EFFECTS ON SEMI-SUPERVISED TRANSFER LEARNING (MAP@.5)

Testset	**1**(RGB**net**)	**1** → **2** (IR **net**)				**2** → **3** (RGB **net**)			
	0.5	0.15	$\boxed{0.35}$ [a]	0.5	0.65	0.15	0.35	0.5	0.65
day	43.7	16.1	22.4	21.8	18.4	40.7 (**-6.92%**)	42.0 (**-3.98%**)	41.6 (**-5.00%**)	35.5 (**-18.8%**)
night	13.4	24.6	31.2	30.1	25.3	20.5 (**+53.2%**)	19.2 (**+43.4%**)	20.3 (**+51.2%**)	16.4 (**+22.2%**)

[a] Only this network has been deployed in the subsequent step (2 → 3).

power. Interpreting these results (3rd vs. 4th row), we indeed observe that the IR object detector is able to gracefully bridge the difference between night and day allowing the retraining of the RGB detector at night.

Subsequent to the last step, the last two rows show the results of the *wormhole learning*. The retrained RGB detector operates significantly better than the original at night (+51.2%) however loses slightly in performance during the day (-5.0%).

We finally want to remark, that this process in general does not put any limit on the size of the new training sets since labels are generated in a semi-supervised way at no labeling cost. This could also overcome the inherent imbalance of categories (Cf. Tab. IV-A), that was present in [31] which in turn leads to poor performance on rare object classes. Also note, that we did not counter-act the imbalance by, e.g., super-sampling rare class objects.

Many theoretical questions arise from this new concept of *wormhole learning*. Fig. 3 already suggests that the concept of *observability* as information that correlates with the task across different domains is still lacking. This becomes a riddle when one has to choose a retrieval threshold for semi-supervised learning. In Tab. IV we present the results obtained generating labels with four different thresholds. Up to the stochastic behavior of the training process it seems that passing more information helps.

V. ANALYSIS OF WORMHOLE LEARNING

We introduce the "wormhole gain" as a possible measure of the performance gain of using wormhole learning. Connected to this, we find nontrivial results; e.g., it turns out that the sensors must be different, but not *too* different.

This analysis hinges on the wormhole gain dependence on:

1) Properties of the two sensors, such as similarity and complementarity.
2) Properties of the two operating domains.

On the other hand, we do not consider the impact of:

1) Finite sample size; we assume to have enough samples to replace the sample average with the expected value.
2) The capability of the learning algorithms and architecture to approximate the function minimizing a given cost function. We thus assume that we can optimize over the real families from which the samples originate.

A. *Formalization of the WHL steps*

We refer to the main sensor as "sensor 1" (the RGB sensor in the experiments) and to the auxiliary sensor as "sensor 2" (IR sensor in the experiments).

Let D be our initial operating envelope, N its expansion, and D + N their union. With $\mathcal{D}$ we denote a dataset and use the exponent to indicate its domain restriction. Thus, $\mathcal{D}^{\mathrm{D}}$ indicates a dataset comprised only of samples from operating envelope D.

We consider the noumenon of a scene sample and label to be drawn from an unobservable distribution $p(\mathrm{Y}, \mathrm{X})$. Recall that in practice, we can observe the scene only through its sensor representation Z. Therefore, similar to [27], the network approximation is interpreted as a proxy for the posterior distribution $f_\theta(\cdot) := q(\mathrm{Y}|\mathrm{Z}, \theta)$. In the following we will adopt $p(\cdot)$ for the unobservable target distributions, and $q(\cdot)$ for the observable network approximations distribution.

Following the wormhole learning process in the Bayesian diagram of Fig. 5, we begin with a detector parametrized by θ_1^{D} trained on the initial dataset $\mathcal{D}^{\mathrm{D}}$ ($A1$ in Fig. 5). Employing the cross-entropy error $H(p, q)$ as the standard cost function for the learning process, we obtain the learned parameters minimizing the loss over the dataset in domain D:

$$\theta_1^{\mathrm{D}} = \arg\min_{\theta} \mathbb{E}_{\mathcal{D}^{\mathrm{D}}} H\left(p_{\overline{\mathrm{Y}}}(\mathrm{Y}|\mathrm{X}), q_\theta(\mathrm{Y}|\mathrm{Z}_1)\right). \qquad (1)$$

As pointed out in [27], the cross-entropy can be decomposed into a sum of terms that individually gauge the information we can learn about the underlying distribution. Loosely speaking, how well $f_{\theta_1^{\mathrm{D}}}$ resembles $p(\mathrm{Y}|\mathrm{X})$ depends on the finite network capacity, the information contained in the dataset, and the optimization process adopted to find θ. In our case, also the sensor representation affects the ability to learn p, since Z might not correlate with the task (see Definition 1).

The first wormhole step considers the sensor domain transfer ($B1$ in Fig. 5). The initial detector acts as "teacher" for the new sensor representation and we find ψ_2^{D} as

$$\psi_2^{\mathrm{D}} = \arg\min_{\psi} \mathbb{E}_{\mathcal{D}^{\mathrm{D}}} H\left(q_{\theta_1^{\mathrm{D}}}(\mathrm{Y}|\mathrm{Z}_1), q_{\psi}(\mathrm{Y}|\mathrm{Z}_2)\right). \tag{2}$$

Subsequently we exploit the new detector $f_{\psi_2^{\mathrm{D}}}$ on the second operating domain N ($A2$ in Fig. 5).

In the second and last step we apply domain transfer to return to the main sensor exploiting samples also from the second operating domain. Thus the final detector parameters $\theta_1^{\mathrm{D+N}}$ minimize the sum of two terms; first the initial cross entropy with the given labels on domain D, and second trying to copy the second sensor on domain N:

$$\begin{aligned}\theta_1^{\mathrm{D+N}} = \arg\min_{\theta} & \mathbb{E}_{\mathcal{D}^{\mathrm{D}}} H\left(p_{\overline{\mathrm{Y}}}(\mathrm{Y}|\mathrm{X}), q_{\theta}(\mathrm{Y}|\mathrm{Z}_1)\right) + \\ & \mathbb{E}_{\mathcal{D}^{\mathrm{N}}} H\left(q_{\psi_2^{\mathrm{D}}}(\mathrm{Y}|\mathrm{Z}_2), q_{\theta}(\mathrm{Y}|\mathrm{Z}_1)\right).\end{aligned} \tag{3}$$

B. Wormhole gain

Hence, for main sensor 1 we define the *wormhole gain* ($\mathsf{WG}_{1\to2}^{\mathrm{D}\to\mathrm{N}}$) of a detector parametrized by $\theta^{\mathrm{D+N}}$ with respect to an initial detector θ^{D} as $\mathsf{WG}_{1\to2}^{\mathrm{D}\to\mathrm{N}}$

$$= \mathbb{E}_{\mathcal{D}^{\mathrm{D+N}}}\left[H\left(p_{\overline{\mathrm{Y}}}, q_{\theta_1^{\mathrm{D}}}(\mathrm{Y}|\mathrm{Z}_1)\right) - H\left(p_{\overline{\mathrm{Y}}}, q_{\theta_1^{\mathrm{D+N}}}(\mathrm{Y}|\mathrm{Z}_1)\right)\right] \tag{4}$$

Before showing some limit cases, we give an intuitive interpretation of the wormhole gain expression. For a given dataset, consider an empirical estimate of the wormhole gain, then from (4) we arrive at

$$\widetilde{\mathsf{WG}} = \sum_{(\mathrm{y},\mathrm{x})\in\mathcal{D}^{\mathrm{D+N}}} p_{\overline{\mathrm{Y}}}(\mathrm{y}|\mathrm{x}) \log\left(\frac{q_{\theta_1^{\mathrm{D+N}}}(\mathrm{y}|\mathrm{z}_1)}{q_{\theta_1^{\mathrm{D}}}(\mathrm{y}|\mathrm{z}_1)}\right). \tag{5}$$

Hence, the gain is $\widetilde{\mathsf{WG}}(k) > 0$ iff the final distribution is closer to the true label distribution than the original distribution.

C. Limit cases

Lemma 1. *If the initial domain is the same as the final domain, WHL learning does not improve performance. If $\mathcal{D}^{\mathrm{D}} = \mathcal{D}^{\mathrm{D+N}}$ then* $\mathsf{WG} \leq 0$.

Proof. By construction of (1) we know that θ_1^{D} achieves the minimum for the first term in (4) because $\mathcal{D}^{\mathrm{D}} = \mathcal{D}^{\mathrm{D+N}}$. Since both terms are non-negative, the wormhole gain is at most zero. Given that $\theta_1^{\mathrm{D+N}}$ as determined in (3) can at best be equal to θ_1^{D}, the wormhole gain is smaller or equal to zero $\mathsf{WG} \leq 0$. □

For the following we define a measure of sensor similarity.

Definition 2. *The similarity index J_{12} is defined as:*

$$J_{1,2} = \frac{I(\mathrm{Z}_1;\mathrm{Z}_2)}{H(\mathrm{Z}_1,\mathrm{Z}_2)}.$$

Note that $J_{1,2}$ (for "Jaccard") is between 0 and 1; $J_{1,2} = 0$ means that the sensors are completely "orthogonal", and $J_{1,2} = 1$ means that the sensors convey exactly the same information, though perhaps represented differently.

Lemma 2. *If the two sensors are equivalent ($J_{1,2} = 1$) and the families of conditional distributions indexed by θ and ψ are equivalent, in the sense that we can put them in a 1-to-1 correspondence, then the wormhole gain* WG *is 0.*

Proof. $J_{1,2} = 1$ implies that the value of Z_1 is fully determined by Z_2. With the assumptions above, following (2) we have $q_{\psi_2^{\mathrm{D}}} = q_{\theta_1^{\mathrm{D}}}$ since this choice minimizes the cross entropy error to zero. Continuing, the second row of (3) is equivalent to (2) and thus zero since both sensors are equivalent. Hence the only way to minimize (3) is to pick $\theta_1^{\mathrm{D+N}} = \theta_1^{\mathrm{D}}$ making the wormhole gain zero. □

Note that in the finite case we would overfit to noisy labels which would lead to $\mathsf{WG} \leq 0$ for similar arguments to Lemma 1.

Lemma 3. *If the two sensors are "orthogonal" ($J(\mathrm{Z}_1;\mathrm{Z}_2) = 0$) then the wormhole gain* WG *is* 0.

Proof. From the "orthogonality" assumption (i) and the data processing inequality (ii) we can deduce that $0 \overset{\text{(i)}}{=} I(\mathrm{Z}_1,\mathrm{Z}_2) \overset{\text{(ii)}}{\geq} I(\mathrm{Y}_1,\mathrm{Z}_2) \overset{\text{(ii)}}{\geq} I(\mathrm{Y}_1,\mathrm{Y}_2) = 0$. Since the output label distributions do not share information, the second term in (3) is independent of the choice of theta. Hence, (3) is again minimized by $\theta_1^{\mathrm{D+N}} = \theta_1^{\mathrm{D}}$. □

Once more, similar to Lemma 2, $\mathsf{WG} \leq 0$ if we drop our idealistic assumptions.

From the previous two lemmas, we have seen that for wormhole learning to be useful, the sensors should be different but not *too* different.

VI. Conclusion

We introduced *Wormhole Learning* as a novel way of leveraging the link between a main and an auxiliary sensor to enlarge the operating domain of the former via semi-supervised learning, providing a simple way of generating "unlimited" labeled data. Crucially, we showed that invariance to undesired changes in data of the auxiliary sensor can be utilized to improve learning outcomes for the first sensor. Through information theoretic analysis of the interplay of the sensors, we offer an understanding of how the characteristics of one type of sensor are related to another and investigated their effect on *wormhole gain* WG. We hope that this will contribute to understanding how heterogeneous sensors relate to each other and advance our insight into how such data may best be put to use.

References

[1] P. Harrison, “The effects of deafness on musical composition.” *Journal of the Royal Society of Medicine*, vol. 81, no. 10, pp. 598–601, 1988.

[2] P. Sachse, U. Beermann, M. Martini, T. Maran, M. Domeier, and M. R. Furtner, ““The world is upside down” – The Innsbruck Goggle Experiments of Theodor Erismann (1883–1961) and Ivo Kohler (1915–1985),” *Cortex*, vol. 92, pp. 222–232, 2017.

[3] H. B. Mitchell, “Introduction,” in *Multi-Sensor Data Fusion*. Springer, 2007, pp. 3–13.

[4] C. Cadena, A. Dick, and I. D. Reid, “Multi-modal Auto-Encoders as Joint Estimators for Robotics Scene Understanding,” in *Rss*, no. June, 2016.

[5] H. Cho, Y.-w. Seo, B. V. K. V. Kumar, and R. R. Rajkumar, “"A Multi-Sensor Fusion System for Moving Object Detection and Tracking in Urban Driving Environments" - Google Search,” in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1836–1843.

[6] R. Calandra, A. Owens, M. Upadhyaya, W. Yuan, J. Lin, E. H. Adelson, and S. Levine, “The Feeling of Success: Does Touch Sensing Help Predict Grasp Outcomes?” *arXiv preprint arXiv:1710.05512*, 2017.

[7] F. Feng, X. Wang, and R. Li, “Cross-modal Retrieval with Correspondence Autoencoder,” in *Proceedings of the ACM International Conference on Multimedia - MM ’14*. ACM, 2014, pp. 7–16. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2647868.2654902

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative Adversarial Networks,” *ArXiv e-prints*, 6 2014.

[9] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, “Learning to Discover Cross-Domain Relations with Generative Adversarial Networks,” *International Conference on Machine Learning (ICML)*, 2017. [Online]. Available: http://arxiv.org/abs/1703.05192

[10] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative Adversarial Text to Image Synthesis,” *PMLR*, 2016. [Online]. Available: http://proceedings.mlr.press/v48/reed16.pdf

[11] M.-Y. Liu and O. Tuzel, “Coupled Generative Adversarial Networks,” in *Advances in Neural Information Processing Systems {(NIPS)}*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 469–477. [Online]. Available: http://arxiv.org/abs/1606.07536

[12] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 10 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5288526/

[13] R. M. Seraj, “Multi-task Learning,” in *Learning to learn*. Springer, 2014, pp. 95–133.

[14] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, “Cross-stitch Networks for Multi-task Learning,” *ArXiv e-prints*, 4 2016.

[15] L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning, S. Legg, and K. Kavukcuoglu, “IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures,” *arXiv preprint arXiv:1802.01561*, 2018. [Online]. Available: http://arxiv.org/abs/1802.01561

[16] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, M. Hughes, and J. Dean, “Google’s Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation,” *arXiv preprint arXiv:1611.04558*, 2016. [Online]. Available: http://arxiv.org/abs/1611.04558

[17] C. Devin, A. Gupta, T. Darrell, P. Abbeel, and S. Levine, “Learning modular neural network policies for multi-task and multi-robot transfer,” in *Proceedings - IEEE International Conference on Robotics and Automation*. IEEE, 5 2017, pp. 2169–2176. [Online]. Available: http://ieeexplore.ieee.org/document/7989250/

[18] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks,” *ArXiv e-prints*, 3 2017.

[19] M.-Y. Liu, T. Breuel, and J. Kautz, “Unsupervised Image-to-Image Translation Networks,” *Advances In Neural Information Processing Systems 31*, no. Nips, 2017. [Online]. Available: http://arxiv.org/abs/1703.00848

[20] C. Chen, Q. Chen, J. Xu, and V. Koltun, “Learning to See in the Dark,” *ArXiv e-prints*, 5 2018. [Online]. Available: http://arxiv.org/abs/1805.01934

[21] Y. Cheng, X. Zhao, K. Huang, and T. Tan, “Semi-supervised learning for rgb-d object recognition,” in *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 2014, pp. 2377–2382.

[22] H. Grabner, C. Leistner, and H. Bischof, “Semi-supervised on-line boosting for robust tracking,” in *European conference on computer vision*. Springer, 2008, pp. 234–247.

[23] X. Zhu, “Semi-supervised learning literature survey,” *Computer Science, University of Wisconsin-Madison*, vol. 2, no. 3, p. 4, 2006.

[24] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski, “Self-supervised Monocular Road Detection in Desert Terrain.” in *Robotics: science and systems*, vol. 38. Philadelphia, 2006.

[25] B. Sofman, E. Lin, J. A. Bagnell, J. Cole, N. Vandapel, and A. Stentz, “Improving robot navigation through self-supervised online learning,” *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 1059–1075, 2006.

[26] L. Pinto and A. Gupta, “Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours,” in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3406–3413.

[27] A. Achille and S. Soatto, “Emergence of Invariance and Disentangling in Deep Representations,” 2017. [Online]. Available: http://arxiv.org/abs/1706.01350

[28] N. Tishby and N. Zaslavsky, “Deep learning and the information bottleneck principle,” in *2015 IEEE Information Theory Workshop, ITW 2015*, 3 2015. [Online]. Available: http://arxiv.org/abs/1503.02406

[29] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, “Deep Variational Information Bottleneck,” 12 2016. [Online]. Available: https://arxiv.org/abs/1612.00410http://arxiv.org/abs/1612.00410

[30] S. Soatto, “Steps Towards a Theory of Visual Information: Active Perception, Signal-to-Symbol Conversion and the Interplay Between Sensing and Control,” 2011. [Online]. Available: http://arxiv.org/abs/1110.2053

[31] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon, “KAIST Multi-Spectral Day/Night Data Set for Autonomous and Assisted Driving,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 934–948, 2018.

[32] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning transferable architectures for scalable image recognition,” *arXiv preprint arXiv:1707.07012*, vol. 2, no. 6, 2017.

[33] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[34] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, “Speed/accuracy trade-offs for modern convolutional object detectors,” *ArXiv e-prints*, 11 2016.

[35] D. A. Van Dyk and X.-L. Meng, “The art of data augmentation,” *Journal of Computational and Graphical Statistics*, vol. 10, no. 1, pp. 1–50, 2001.

[36] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes (VOC) challenge,” *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[37] Tzutalin, “LabelImg,” 2015. [Online]. Available: https://github.com/tzutalin/labelImg