# Real-Time Ultrasound Transducer Localization in Fluoroscopy Images by Transfer Learning from Synthetic Training Data

Tobias Heimann[a,*], Peter Mountney[b], Matthias John[c], Razvan Ionasec[b,1]

[a]*Siemens AG, Corporate Technology, Imaging and Computer Vision, Erlangen, Germany*
[b]*Siemens Corporation, Corporate Technology, Imaging and Computer Vision, Princeton, NJ, USA*
[c]*Siemens AG, Healthcare Sector, Forchheim, Germany*

**Abstract**

The fusion of image data from trans-esophageal echography (TEE) and X-ray fluoroscopy is attracting increasing interest in minimally-invasive treatment of structural heart disease. In order to calculate the needed transformation between both imaging systems, we employ a discriminative learning (DL) based approach to localize the TEE transducer in X-ray images. The successful application of DL methods is strongly dependent on the available training data, which entails three challenges: 1) the transducer can move with six degrees of freedom meaning it requires a large number of images to represent its appearance, 2) manual labeling is time consuming, and 3) manual labeling has inherent errors.

This paper proposes to generate the required training data automatically from a single volumetric image of the transducer. In order to adapt this system to real X-ray data, we use unlabeled fluoroscopy images to estimate differences in feature space density and correct covariate shift by instance weighting. Two approaches for instance weighting, probabilistic classification and Kullback-Leibler importance estimation (KLIEP), are evaluated for different stages of the proposed DL pipeline. An analysis on more than 1900 images reveals that our approach reduces detection failures from 7.3% in cross validation on the test set to zero and improves the localization error from 1.5 to 0.8 mm. Due to the automatic generation of training data, the proposed system is highly flexible and can be adapted to any medical device with minimal efforts.

*Keywords:* Transfer learning, Domain adaptation, Object Localization, Fluoroscopy, Ultrasound

## 1. Introduction

Catheter-based procedures such as trans-aortic valve implantation (TAVI) or paravalvular leak closure are gaining increasing importance for the treatment of structural heart disease. The inherent challenge for the cardiac interventionalist is to infer the exact position of the catheter relative to the tissue from the available imaging information. X-ray fluoroscopy is the dominant imaging modality for these interventions, increasingly supported by 3D trans-esophageal echography (TEE) (Gao et al., 2012). Both modalities show complementary information, but in clinical practice they are controlled and displayed completely independently from each other.

Recently, image fusion has been proposed to combine both modalities and to provide the cardiac interventionalist with a better overview of the *in situ* conditions. The co-registration can be accomplished by means of electromagnetic (EM) tracking (Jain et al., 2009), but this approach requires EM tracking hardware to be attached to the transducer and is sensitive to EM field distortions. In Ma et al. (2010), authors present a feasibility study with a robotic arm for tracking a trans-thoracic echo probe. Apart from the difficulties of extending this system to TEE probes, the robotic hardware requirements severely limit the practical applicability of this approach. Alternatively, the pose of the transducer can be estimated from its appearance in the X-ray images, either directly (Gao et al., 2012; Mountney et al., 2012) or supported by fiducial markers attached to the probe head (Lang et al., 2012). Since the former approach does not require additional hardware, it is advantageous for integration into the clinical workflow, albeit more challenging to implement.

While 2D-3D registration (Gao et al., 2012) yields accurate results, it has a limited capture range of $< 10$ mm, requiring a manual initialization every time a new fluoroscopy sequence is acquired. Discriminative learning (DL) (Mountney et al., 2012) can locate the TEE probe everywhere in the image, but its performance is strongly dependent on quantity and quality of the available training data. In the medical domain, data is generally difficult to acquire, and the required manual labeling is an extremely

*Corresponding author at: Siemens AG, Corporate Technology, Imaging and Computer Vision, Erlangen, Germany. Tel.: +49 9131 724607

*Email addresses:* tobias.heimann@siemens.com (Tobias Heimann), peter.mountney@siemens.com (Peter Mountney), matthias.mj.john@siemens.com (Matthias John), razvan.ionasec@siemens.com (Razvan Ionasec)

[1]Present address: Siemens AG, Healthcare Sector, Forchheim, Germany

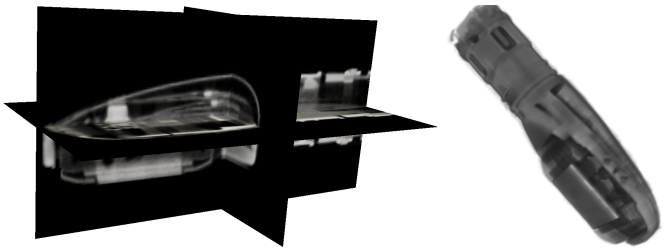

Figure 1: 3D visualization of the processed C-arm CT volume of an X7-2t 3D TEE transducer (Philips, The Netherlands).

tedious and time-consuming task. Moreover, trained operators cannot reproducibly annotate images with perfect accuracy, and every variation in ground truth will decrease the performance of the resulting DL system.

In this paper, we propose a novel approach for training a DL system based on *in silico* training data that can be generated automatically in large quantities with perfectly accurate labels. Since synthetic image generation cannot faithfully model all aspects of *in vivo* fluoroscopy data, the DL system must be adapted. For this purpose we employ unsupervised domain adaptation, a technique which has been widely used in speech processing and has recently gained attention in the computer vision community (Margolis, 2011; Beijbom, 2012). In particular, we show how unlabeled data from the target domain (i.e. *in vivo* images) can be used to improve the performance of object localization beyond what is achievable with semi-supervised learning (Zhu, 2008). We apply our approach to the estimation of in-plane parameters of a TEE probe in fluoroscopy images, i.e. 2D position, in-plane orientation, and scale.

This article is an extended version of Heimann et al. (2013); it explains the methodology in more detail and adds a number of new experiments to the domain adaptation. While based on the same image data, this new version uses updated, more accurate annotations for the *in silico* images, which leads to slightly different results in the evaluation. We start with presenting the basic learning method in the next section and explain our adaptation approach afterwards.

## 2. Learning from Synthetic Data

### *2.1. Generation of in silico Images*

The synthetic training data is based on digitally reconstructed radiographs, which approximate X-ray images from computed tomography (CT) volumes. The source is a high-resolution (0.18 mm/voxel) isotropic C-arm CT of the TEE transducer, which was aligned to the image axes and cropped to contain only the probe head. A binary mask of the transducer was prepared and multiplied with the original volume to remove streak artifacts in the surrounding air. Figure 1 shows the final transducer volume in three-plane view and volume visualization.

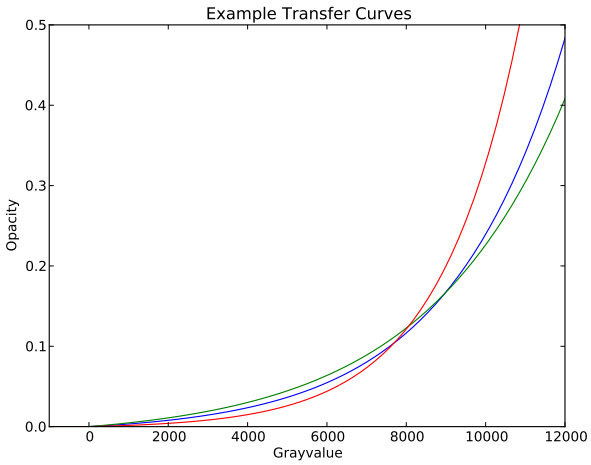


Figure 2: Examples for different transfer curves used for generating *in silico* images.

For each synthetic image, we set up a virtual scene that represents a realistic C-arm geometry. The camera is located 120 cm away from the image plane and features a view angle between 6.5 and 11 degrees, simulating different zoom modes of the C-arm. The 3D position and three Euler angles of the virtual transducer are randomized with the constraints that a) the probe is located at a distance between 33 and 47 cm away from the image plane, b) the projected probe is completely inside the image frame, and c) the probe head is oriented in inferior direction. The flexible tube to which the probe is attached is modeled by a 3D spline originating from a random position at the upper image boundary. Along this spline, a collection of rings is positioned in regular pattern. This is consistent with *in vivo* images captured during structural heart procedures.

2D projections are generated using a composite ray-caster, i.e. every pixel is assigned the sum of all values along the respective ray through the volume. Key to generating realistic-looking images is the transfer function used to calculate the opacities along the ray. Based on the appearance of *in vivo* images, we chose an exponential transfer function with randomized parameters in order to generate sequences with slightly varying appearance and contrast. A gray value $x > 0$ in the TEE volume maps to opacity $\alpha(x)$ as follows:

$$\alpha(x) = c_0 \left( \exp\left(\frac{x}{c_1}\right) - 1 \right) / \left( \exp\left(\frac{7500}{c_1}\right) - 1 \right) \quad (1)$$

with $c_0 \in [0.08, 0.12]$ setting the opacity for gray value 7500 and $c_1 \in [2200, 3800]$ setting the contrast as randomized parameters. Figure 2 shows some example curves for different values of $c_0, c_1$.

As background, we used 12 cardiac fluoroscopy sequences without transducer and combined them with the generated ray-caster images by additive blending. Annotations were created automatically by storing the 2D position of

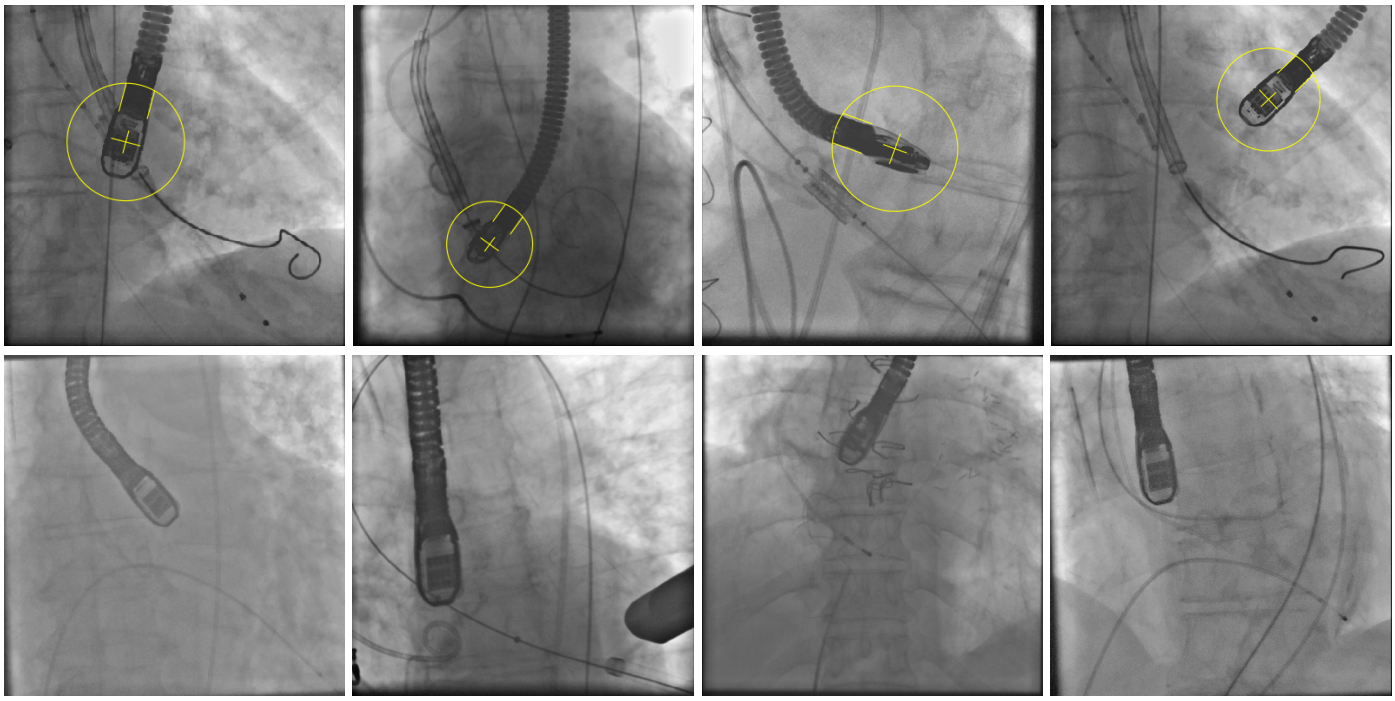

Figure 3: A selection of generated *in silico* images with automatic annotations (top row) and *in vivo* fluoroscopy images (bottom row).

a fixed point in the center of the transducer together with the respective Euler angles and the probe scale. Since the apparent size in the projected 2D image varies with the rotation angles, scale is measured as the width of uppermost, circular part of the transducer which connects to the flexible tube. Figure 3 gives an impression of the look of the generated images compared to *in vivo* data.

### *2.2. Transducer Localization by Discriminative Learning*

Following the marginal space learning approach (Zheng et al., 2008), transducer localization is performed in several stages by a pipeline of three discriminative classifiers. The first classifier $\Phi$ employs Haar-like features $x_H$ (Viola and Jones, 2004) to determine the 2D position of the probe in images rescaled to 1 mm isotropic pixel spacing. All pixels closer than 1 mm to the reference annotation are labeled as $y = Y^+$, all others as $y = Y^-$. During detection, the 50 candidates with the highest classifier output $\hat{p}_\Phi(y = Y^+|x_H)$ are passed on to the in-plane orientation detector $\Theta$.

$\Theta$ is based on a 4×4 grid of steerable features $x_S$ (Zheng et al., 2008) calculated at 0.25 mm isotropic resolution. Possible angles of the transducer are discretized into 6° steps, and all correctly positioned samples deviating $< 4°$ from the annotated angle are labeled as $Y^+$. For test images, the 50 candidates with the highest $\hat{p}_\Theta(y = Y^+|x_S)$ are passed on to scale detector $\Psi$.

$\Psi$ is again based on steerable features $x_S$ with 0.25 mm spacing, but uses a much finer $32 \times 32$ grid. The observed TEE scales of 7.5–11 mm are discretized into a set of 8 hypotheses, corresponding to feature window sizes from 30–44 mm. Lastly, the 50 highest-ranked candidates are combined by weighted averaging according to their respective $\hat{p}_\Psi(y = Y^+|x_S)$ and produce the final output. All classifiers of the pipeline are implemented as probabilistic boosting trees (PBTs) (Tu, 2005), which combine high computational efficiency with competitive accuracy.

Some parameters of this pipeline, like the grid sizes for steerable filters, have been optimized empirically for this specific application. Others, as the number of candidates passed on to the next level or the image resolution at different stages, are not that critical for the performance of the system and have been set according to our experience with other applications.

## 3. Transfer Learning

A fundamental assumption in machine learning is that training and test data stem from the same distribution. In our approach, however, the training data originates from the *in silico* source domain $S$, while the test data comes from the *in vivo* target domain $T$. Consequently, the above assumption may not hold, in which case the classifiers would work along non-optimal decision boundaries.

Formally, let $x$ represent a feature vector for a sample and $y \in [Y^+, Y^-]$ its label, then the joint probability distribution $P(y, x)$ should be identical for source and target domain. In our case, we know that the marginalized label probabilities are equal, i.e. $P_S(y) = P_T(y)$, since images from both domains show exactly one transducer. Moreover, given a certain feature vector, the question if the corresponding image region shows a probe can also be decided without knowing its domain, which makes it

reasonably safe to assume that $P_S(y|x) = P_T(y|x)$. However, the distribution of feature vectors in both domains is probably different, i.e. $P_S(x) \neq P_T(x)$, which leads to a situation called covariate shift (Shimodaira, 2000).

The effect of covariate shift on a classifier is illustrated intelligibly by Yamada et al. (2012), among others: Classifiers typically model class boundaries by a number of parameters and, during training, select optimal values for these parameters based on some error minimization in the source domain. In order to prevent over-fitting, the number of parameters and thus the possibilities for class boundaries are limited, which means that the learned model performs better in denser regions of the training data. If the target domain is only sampled sparsely during training, considerable errors can occur when applying the classifier there.

In machine learning, approaches to adopt existing classifiers to new tasks or new domains are broadly labeled as transfer learning. The covariate shift problem encountered in our case falls into the category of transductive transfer learning or domain adaptation (Pan and Yang, 2010). In the following, we present the general idea of the approach and how we can use it for object localization.

### *3.1. Learning under Covariate Shift*

As described by Shimodaira (2000), a classifier can be adapted to different training and test distributions by minimizing its loss function. This is accomplished by assigning each training sample an instance weight according to the ratio of joint probabilities of target and source domain. Under covariate shift, this ratio simplifies to:

$$\frac{P_T(y,x)}{P_S(y,x)} = \frac{P_T(x)P_T(y|x)}{P_S(x)P_S(y|x)} = \frac{P_T(x)}{P_S(x)} \tag{2}$$

Conveniently, this formulation does not include any labels $y$, i.e. no annotations are required for the target domain in order to adapt the classifier.

The challenge lies in estimating the required density ratio (Sugiyama et al., 2010b): Estimating a continuous density function from samples is non-trivial in itself, but dividing by an estimated density can magnify the occurring errors. For this reason, a number of approaches to estimate the required density ratio directly have been developed (Sugiyama et al., 2010b). One of these is the probabilistic classification approach, in which a classifier is trained to differentiate between samples $x_S \in S$ and $x_T \in T$. Sugiyama et al. (2010b) present logistic regression (Hastie et al., 2009) as a suitable classifier for this task. During training, all $x_S$ are assigned to $y = 1$ and all $x_T$ to $y = 0$. The density ratio can then be estimated using classifier output $\hat{p}$ by:

$$\frac{P_T(x)}{P_S(x)} = \frac{1}{\hat{p}(y=1|x)} - 1 \tag{3}$$

According to a theoretical analysis in Kanamori et al. (2010), this approach is optimal in case that the joint probabilities of source and target domain are members of the exponential family. In practice, this is rarely the case, and the so-called ratio matching approach should deliver better results (Sugiyama et al., 2010b).

The core idea of ratio matching is to construct a parametric density ratio model and match this to the true density ratio. Since the latter is not available, a divergence method is used to measure the error of the model. A recent method following this approach is the Kullback-Leibler importance estimation procedure (KLIEP) (Sugiyama et al., 2008). It can be used with a Gaussian kernel model for the density ratio and determine the optimal variance for the kernels during the estimation. For a mathematical derivation of the method, we refer the reader to Sugiyama et al. (2008). The authors also provide Matlab code for the algorithm[2].

### *3.2. Instance Weighting for Object Localization*

With logistic regression and KLIEP, we have two viable approaches to estimate weights for training our classifiers $\Phi$, $\Theta$, and $\Psi$. However, while instance weighting has already been employed for a number of different tasks (Margolis, 2011), its application to object localization raises two important questions: Which samples should be used to estimate the feature density ratio, and should positive and negative samples be treated equally for weighting?

Using all available samples would require extracting feature vectors for every pixel in every available image multiple times (for different orientation and scale hypotheses). Not only would this result in the impractical amount of $10^{12}$ feature vectors, but it would also lead to highly unbalanced class labels $Y^+$ and $Y^-$. Moreover, as we use a relatively small number of background sequences to generate the *in silico* data, features for $Y^-$ are repeating in the source domain. In summary, this would lead to background samples $Y^-$ completely dominating the density ratio estimation, although it is the appearance of the transducer (labels $Y^+$) which should ideally drive the domain adaptation.

We propose a two-step approach to solve this problem. In order to generate a subset of samples, we employ a DL pipeline trained on *in silico* data $S_1$ to localize the transducer in another set of synthetic images $S_2$ and unlabeled *in vivo* data $U$. As even an average DL system will detect the transducer with reasonable accuracy on the majority of images, this step effectively reverses the class imbalance in favor of positive samples $Y^+$. Feature vectors for the generated samples are normalized to zero mean and unit variance over the entire set. They are used as input to either the logistic regression or KLIEP. As the quality of the density ratio estimation may vary, we relax instance weight $w$ as suggested by Shimodaira (2000):

$$w(x) = \left(\frac{P_T(x)}{P_S(x)}\right)^c \tag{4}$$

[2] http://sugiyama-www.cs.titech.ac.jp/ sugi/software/KLIEP/

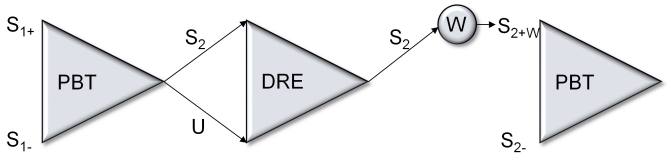


Figure 4: Overview of the proposed approach for domain adaptation. In order to bias instance weighting towards positive samples, a probabilistic boosting tree (PBT) classifier is trained on synthetic image set $S_1$. Density ratio estimation (DRE) is then used to calculate instance weights for $S_2$. Finally, the domain adapted PBT classifier is trained with weighted positive samples and standard negative samples of $S_2$.

with $c \in [0, 1]$ as regularization parameter. In this study, we set $c = 0.5$.

This procedure provides instance weights for all generated samples of $S_2$, which are assigned as weights to the corresponding synthetic images. Ideally, these weights should be high for images with similar appearance to *in vivo* data, and low for less similar ones. Please note that, when the domain adapted classifier is trained on $S_2$, it uses the ground truth annotations for $Y^+$ and not the previously generated samples. In addition, since the instance weights were generated with a bias on positive samples, they are only used for samples $Y^+$, while background samples $Y^-$ remain unweighted. Figure 4 summarizes this approach graphically.

## 4. Experiments and Results

### *4.1. Image Data*

Image data originates from two clinical centers and was mostly acquired during standard TAVI procedures. Both centers used an Artis Zeego C-arm system (Siemens AG, Germany) for acquisition of fluoroscopy and an X7-2t 3D transducer (Philips, The Netherlands) for acquisition of TEE. In order to estimate the physical resolution of each fluoroscopy sequence, the pixel spacing of the fluoroscopic detector was divided by the radiologic magnification factor, which accounts for the projection geometry of the C-arm. In order to prevent problems with local feature calculation, we excluded approx. 25% of all frames in which the transducer was too close to the image boundaries. In prospective clinical application, the X-ray window could always be chosen to include the probe entirely, i.e. this data exclusion does not limit the applicability of the proposed approach.

In the end, we used 68 sequences from 22 patients for our study, comprising 6280 frames in total. For 37 sequences comprising 1913 frames, the probe head was annotated manually by two engineering students (who distributed the complete workload between themselves). We denote this set of annotated *in vivo* images as $T_L$, while the remaining unlabeled 4367 frames are denoted as $T_U$. Finally, using the method from Sec. 2.1, we generated two sets $S_1, S_2$ containing 10,000 *in silico* images each. These two synthetic image sets are required for the proposed domain adaptation approach, as described in Sec. 3.2.

Table 1: Mean errors for manual annotations on 220 frames.

| | Position Error | Orientation Error | Scale Error |
|---|---|---|---|
| Student 1 | 0.4±0.2 mm | 0.7±0.5° | 0.5±0.2 % |
| Student 2 | 0.4±0.2 mm | 0.7±0.6° | 8.5±0.2 % |

Table 2: Area-under-curve values and changes relative to baseline for different instance weights (IW) on the position detector.

| System | AUC | Change to BL |
|---|---|---|
| Baseline | 88.8 | |
| IW LR70 | 95.2 | +7.2 % |
| IW LR10 | 86.3 | -2.9 % |
| IW KLIEP70 | 80.7 | -9.1 % |
| IW KLIEP10 | 79.7 | -10.2 % |

### *4.2. Manual Annotation*

Before generating synthetic image sets $S_1, S_2$, we conducted a small annotation study in which the two students annotated 220 *in silico* frames independent from each other. For each frame, the students had to place an oriented rectangle over the probe head according to a fixed protocol. One side of the rectangle marks the proximal and one side the distal end of the probe head, while the lengths of these sides represent the width of the probe shaft.

As described in Sec. 2.1, we require a fixed 3D position in the probe volume that can be projected to DRRs for the automatic labeling of 2D location. We selected this 3D position as the point that minimized the average error to the center of the manually annotated rectangles in the 220 frames of the annotation study. Finally, we compared the annotations of both students to the thus generated ground truth (see Table 1). As can be seen from the results, manual annotation is reproducible to a satisfying level, except that Student 2 consistently annotated a larger width of the probe head. We went over the protocol again with him and made sure he annotated the width correctly. After this, both students proceeded with the annotation of the *in vivo* images.

### *4.3. Logistic Regression vs. KLIEP*

In Sec. 3.1, we presented two different approaches for generating instance weights: Logistic regression (LR) and KLIEP. In order to determine which method is better suited for our application, we evaluated the performance of different weighting schemes on the position detector $\Phi$. An unweighted detector $\Phi_0$ trained on $S_2$ serves as baseline for

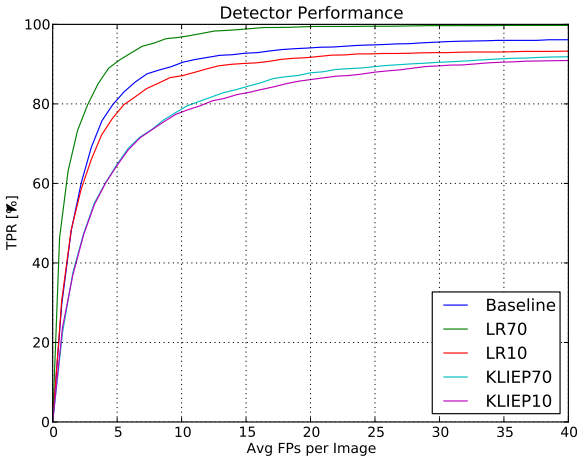


Figure 5: True positive rate (TPR) vs. average number of false positives (FPs) for different instance weights on the position detector.

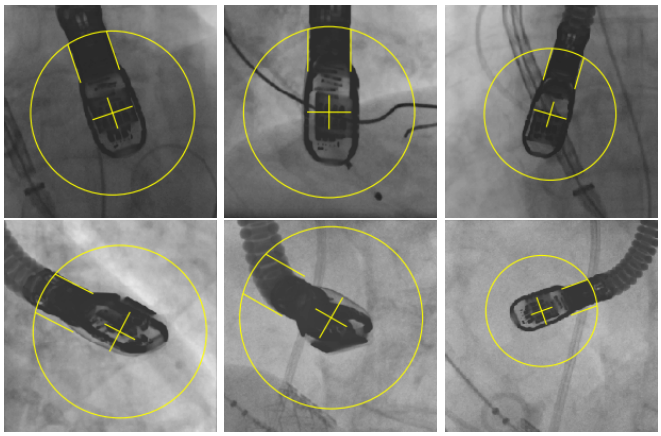

Figure 6: A selection of *in silico* training samples that received high instance weights (top) and low instance weights (bottom) for the position detector. Automatic annotations are visualized as yellow overlays. All weights were estimated by logistic regression on the full feature set.

the experiment. It was set up as three-level cascade with 10, 20, and 40 Haar-like features per level, respectively.

For the domain-adapted classifiers, a full PBT pipeline ($\Phi \Rightarrow \Theta \Rightarrow \Psi$) was trained on $S_1$ and used to generate TEE probe samples from $S_2$ and $T_U$, as described in Sec. 3.2. Four different sets of instance weights (IW) were estimated from these samples:

- IW LR70: based on the full feature set and logistic regression
- IW LR10: based the 10 most discriminating features of the first cascade level
- IW KLIEP70: based the full feature set and KLIEP
- IW KLIEP10: based on the 10 most discriminating features

The four domain-adapted classifiers $\Phi_A^i, i \in [1 \dots 4]$ resulting from training on $S_2$ with the respective positive weights were evaluated on $T_L$. All detected candidates with a position error $< 1$ mm were counted as true positives. Plotting these counts against the average number of false positives results in the curves shown in Fig. 5. The corresponding areas under the curve (AUCs) are given in Table 2. As can be seen, the only weighting approach that improves detection results is logistic regression on the full feature set. Some examples for samples that obtained very high and low weights with this method are shown in Fig. 6. Reducing the number of features for density estimation deteriorates the performance of the resulting classifier. Weights determined by KLIEP yield the worst performance for our application.

### *4.4. Selecting the Stages for Domain Adaptation*

According to the outcome of the weighting scheme evaluation, logistic regression on the full feature set was chosen to estimate instance weights. The second set of experiments was conducted to analyze in how far the encouraging results reached for domain adaptation of the position detector carry over to the other stages of the detector pipeline. As baseline system ($\Phi_0 \Rightarrow \Theta_0 \Rightarrow \Psi_0$), we trained the complete pipeline presented in Sec. 2.2 on $S_2$. Orientation detector $\Theta_0$ was set up as three-level tree with 105 steerable features in total, and scale detector $\Psi_0$ as three-level tree with 190 steerable features.

For each image of $S_2$, the same TEE probe sample as in the previous section was used to calculate instance weights for both detectors, using their respective steerable feature sets. These weights were used as input for the domain-adapted classifiers $\Theta_A$ and $\Psi_A$. In order to select the stages for domain adaptation (DA), three different pipelines were assembled:

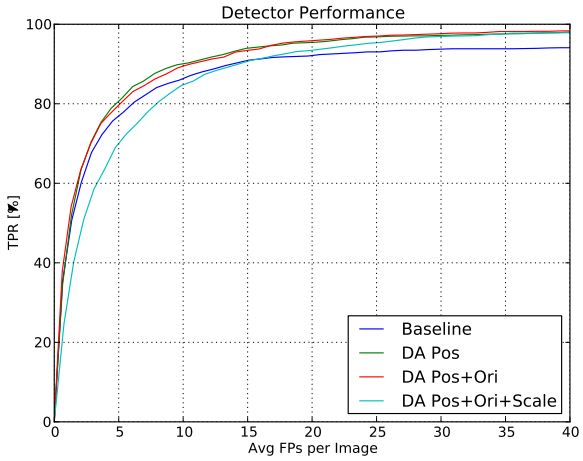


Figure 7: True positive rate (TPR) vs. average number of false positives (FPs) for detection pipelines employing domain adaptation at different stages.

Table 3: Area-under-curve values for detection pipelines employing domain adaptation at different stages. All changes are given relative to baseline.

| System | AUC | Change to BL |
|---|---|---|
| Baseline | 86.8 | |
| DA Pos | 90.2 | +3.9 % |
| DA Pos+Ori | 90.3 | +4.0 % |
| DA Pos+Ori+Scale | 86.4 | -0.4 % |

- DA Pos: uses instance weighting just for the position detector ($\Phi_A \Rightarrow \Theta_0 \Rightarrow \Psi_0$)
- DA Pos+Ori: uses instance weighting for position and orientation detectors ($\Phi_A \Rightarrow \Theta_A \Rightarrow \Psi_0$)
- DA Pos+Ori+Scale: uses instance weighting for all stages ($\Phi_A \Rightarrow \Theta_A \Rightarrow \Psi_A$)

As before, all systems were evaluated on image set $T_L$. For a detailed analysis of each system, we looked at the detected candidates before the final averaging step and counted a true positive if one of the candidates had a position error $< 1$ mm, an orientation error $< 4°$, and a scale error $< 0.75$ mm (scale measured as width of the tube). These thresholds were chosen so that for each parameter, the maximum allowable deviation will result in an average movement of 1 mm, calculated over the area of the feature window.

Figure 7 shows the resulting performance curves and Table 3 the corresponding areas under the curve. As can be seen, domain adaptation on the position detector has the largest impact with an increase of 3.9% AUC relative to the baseline system. Domain adaptation on the orientation detector brings only slight additional improvements (+4.0% AUC relative to baseline), while trying to adapt the scale detector actually deteriorates the results (-0.4% AUC relative to baseline).

### *4.5. Evaluation of Robustness and Accuracy*

Following the results of the previous section, the system with instance weighting for position and orientation detectors was selected for the final evaluation of robustness and accuracy. In order to assess its performance relative to the state of the art, it was compared to a number of alternative systems:

- *in vivo* Reference: trained directly on $T_L$ without any synthetic data (using three-fold cross-validation for evaluation)
- *in silico* Baseline (from Sec. 4.4): trained exclusively on synthetic images $S_2$
- Self Training: Following a popular approach in semi-supervised learning (Zhu, 2008), the samples drawn from $T_U$ (as described in Sec. 3.2) are used to enlarge the synthetic training set and to generate another unweighted system from $S_2 \cup T_U$.
- Train on Test Data: To evaluate the commonly best case in machine learning, this system is trained directly on the test set $T_L$.

For each system, the final output of the pipeline (after candidates are merged) was compared to the reference labels. In case the output was located outside the annotated probe area (circles in Figs. 3 and 6), the localization was counted as failure. For successful detections, average position, orientation and scale errors were computed. The complete results are displayed in Table 4.

The entire detection pipeline runs in less than 25 ms per frame on an Intel i7 Quadcore CPU with 2.2 Ghz. This time holds true for all tested systems, as the differences lie only in the respective training procedures. During minimally invasive interventions, fluoroscopy sequences are typically acquired with 7 to 15 frames per second. Therefore, the presented approach is completely real-time capable.

### *4.6. Analysis of Feature Sets*

The unexpectedly low performance of the pipeline "DA Pos+Ori+Scale" prompted us to take a closer look at the different feature sets that the instance weights are based on. Each individual feature (Haar-like or steerable) evaluates a certain region of the complete feature window. During training of a PBT, the features which best discriminate between positive and negative samples are included in the respective stage of the tree or cascade (Tu, 2005). Since sample weights are included in the calculation of the discriminative power, it is possible that the domain-adapted classifiers use different feature sets than the baseline classifiers. Figure 8 visualizes the selected features for position, orientation, and scale classifiers before and after domain adaptation. To quantify the differences, we calculated the Dice coefficient for overlap of corresponding feature sets before and after domain adaptation. These values are listed in Table 5.

Since the overlap is less than 50% for all three detectors, we decided to analyze the sensitivity of instance weights with regard to different feature sets. For this, we re-calculated the respective instance weights based on the feature sets after domain adaptation and calculated their correlation to the original weights. Due to the distinctively non-Gaussian distribution of weights, Spearman's correlation coefficient $\rho$ was used for this purpose. Results are again shown in Table 5.

## 5. Discussion

Our results clearly demonstrate the dependency of DL systems on the available training data. The reference system in our experiments, although trained on the same domain as the test data, yields the worst overall results. The *in silico* system can compensate the difference between

Table 4: Mean errors with standard deviation for successful detections.

| | Failed Detections | Position Error | Orientation Error | Scale Error |
|---|---|---|---|---|
| *in vivo* Reference | 7.34 % | 1.5±2.5 mm | 3.2±5.4° | 3.8±3.0 % |
| *in silico* Baseline | 3.19 % | 0.9±0.6 mm | 1.7±1.3° | 3.1±2.0 % |
| Self Training | 0.42 % | 0.8±0.9 mm | 1.5±1.3° | 4.3±3.0 % |
| Train on Test Data | 0.37 % | 0.7±0.8 mm | 1.5±1.3° | 2.6±2.5 % |
| Domain Adaptation | 0.00 % | 0.8±0.5 mm | 1.4±1.1° | 3.2±2.3 % |

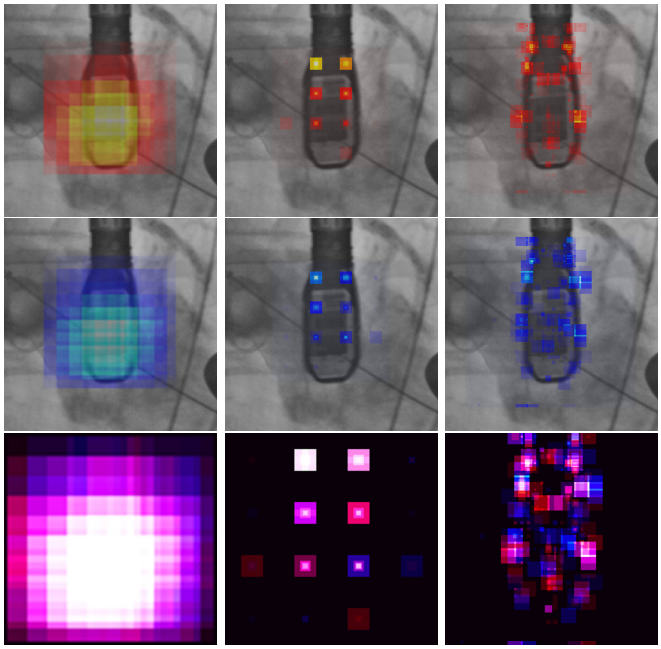

Figure 8: Visualization of feature sets for position detector (left column), orientation detector (middle column), and scale detector (right column). The first row shows features for the baseline versions in red, the second row features for the domain-adapted versions in blue. The shades of each color from black to white indicate the number of features relying on the respective areas. The last row visualizes the differences between both feature sets (with white and magenta encoding overlapping regions).

source and target domain by an eight times larger training set with perfectly placed labels and reduces the number of failed detections to less than a half, while at the same time improving on all errors. Regarding the combination of synthetic data with unlabeled *in vivo* images, self training is very straight-forward to implement and yields excellent results: Errors are comparable to the *in silico* system (slight improvements for position and orientation with approx. 40% worse scale estimation), but the misdetections are reduced considerably further down to 6% of the reference system.

Given these strong results and the problems of replicating the promising results of the instance-weighted position detector at later stages of the pipeline (especially scale), we were surprised that domain adaptation was still able

Table 5: Change of feature sets by domain adaptation (Dice overlap) and influence on instance weights (Spearman's $\rho$).

| Detector | Dice overlap | Spearman's correlation |
|---|---|---|
| Position | 10.5 % | 88.8 % |
| Orientation | 43.3 % | 86.8 % |
| Scale | 14.4 % | 54.1 % |

to improve upon these numbers. Reducing misdetections by an additional half percent might not seem like much, but in general, eliminating failures becomes more difficult with higher base performance, and domain adaptation reduces the number of failures to zero in this study. Its success is based on up-weighting training samples that appear similarly in the target domain and down-weighting less common samples with e.g. very high contrast or large rotations (see Fig. 6). Obviously, generating *in silico* data with more realistic parameters from the start would have a similar effect, but – as for most applications – the true distribution of parameters in real-world data is not known.

Since the largest difference between source and target domain appears in the feature set of the position detector (which has to cope with different orientations and scales), this stage of the pipeline can benefit most from domain adaptation. While the orientation detector still improves with domain adaptation, instance weighting completely failed for the scale detector. Based on the results from Sec. 4.6, we believe this is due to the inherent feature selection, which produces different feature sets for different instance weights. While the position detector also changes almost 90% of its original features after weighting, its new feature set covers essentially the same area as before, which leads to a high correlation of the corresponding weights. The steerable features of the scale detector, however, cover different image regions after weighting, and the resulting weights correlate only mildly with the original values. In future work, enforcing the same feature set for the domain-adapted PBTs as for the original detectors will be an interesting experiment.

Regarding the different methods we evaluated to esti-

mate the required instance weights for domain adaptation in Sec. 4.3, it seems like using a lower-dimensional subset of features does not lead to usable weights. Both logistic regression and KLIEP yielded worse results than no weighting when only 10 features were used. While logistic regression delivered convincing results with the full set of 70 features, KLIEP improved only slightly and stayed below the baseline. We suspect that 70 features are already too much for KLIEP to work reliably – as reported in Sugiyama et al. (2010a), the standard KLIEP approach performs poorly when the dimensionality of the data is high.

The detection pipeline we presented in this work only estimates the in-plane parameters of the TEE probe. For an information fusion between TEE and fluoroscopy in the final clinical application, out-of-plane parameters (two additional rotations) are also required. These missing parameters can be estimated by subsequent 2D-3D registration (Gao et al., 2012) or template-matching (Mountney et al., 2012), which will be initialized with the values from the presented pipeline. As such, the most important parameter for the presented system is the robustness of detection. In case the probe is misdetected on a frame, the subsequent stage will not be able to recover from this error. Accuracy is important mainly because a more accurate initialization allows reducing the search range for the following stage, which will result in faster run-time. According to initial experiments using the template matching approach from (Mountney et al., 2012), the obtained accuracy is sufficient to allow for a real-time detection of the complete 3D pose.

## 6. Conclusions

In summary, we believe the combination of automatically generated data and unlabeled real-world images to be a highly promising approach for training DL systems. It resolves the need for thousands of annotated training samples, which is one of the main bottlenecks of machine learning in the medical domain. Moreover, the ability to create large quantities of training data for any X-ray imageable device (e.g. implants or new transducers) within hours offers unmatched flexibility.

The presented approach for TEE localization is extremely robust, highly accurate, and real-time capable, properties which make it ideal for initializing subsequent 2D-3D registration (Gao et al., 2012) or template-matching (Mountney et al., 2012) approaches. This combination will allow to gain the complete 3D pose of the transducer fully automatically, which will facilitate the interventional fusion of TEE and fluoroscopy images (as exemplified in Fig. 9) and hopefully contribute to making minimally-invasive procedures for structural heart disease even safer.

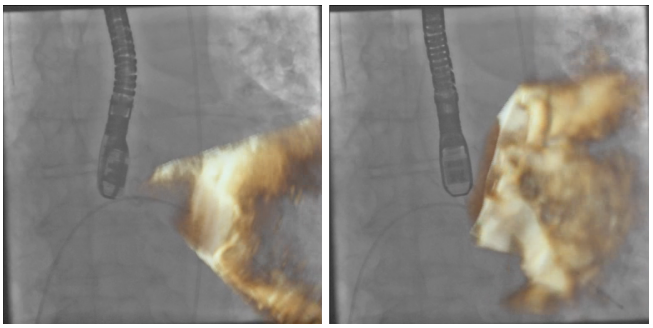

Figure 9: Potential clinical application of interventional image fusion: The 3D TEE images can be shown together with the fluoroscopy data and from the same viewpoint. This enables soft tissue information (from ultrasound) and tool location (from fluoroscopy) to be visualized in the same coordinate system.

## References


Beijbom, O., 2012. Domain Adaptation for Computer Vision Applications. Technical Report. University of California, San Diego.

Gao, G., Penney, G., Ma, Y., Gogin, N., Cathier, P., Arujuna, A., Morton, G., Caulfield, D., Gill, J., Rinaldi, C.A., Hancock, J., Redwood, S., Thomas, M., Razavi, R., Gijsbers, G., Rhode, K., 2012. Registration of 3D trans-esophageal echocardiography to X-ray fluoroscopy using image-based probe tracking. Med Image Anal 16, 38–49.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd edition ed., Springer.

Heimann, T., Mountney, P., John, M., Ionasec, R.I., 2013. Learning without labeling: Domain adaptation for ultrasound transducer localization, in: Proc MICCAI, pp. 49–56.

Jain, A., Gutierrez, L., Stanton, D., 2009. 3D TEE registration with X-ray fluoroscopy for interventional cardiac applications, in: Proc. Functional Imaging and Modeling of the Heart (FIMH), Springer. pp. 321–329.

Kanamori, T., Suzuki, T., Sugiyama, M., 2010. Theoretical analysis of density ratio estimation. IEICE Transactions on Fundamentals of Electronics, Communication and Computer Science E93-A, 787–798.

Lang, P., Seslija, P., Chu, M.W.A., Bainbridge, D., Guiraudon, G.M., Jones, D.L., Peters, T.M., 2012. US - fluoroscopy registration for transcatheter aortic valve implantation. IEEE Trans Biomed Eng 59, 1444–1453.

Ma, Y., Penney, G.P., Bos, D., Frissen, P., Rinaldi, C.A., Razavi, R., Rhode, K.S., 2010. Hybrid echo and X-ray image guidance for cardiac catheterization procedures by using a robotic arm: a feasibility study. Physics in Medicine and Biology 55, 371–382.

Margolis, A., 2011. A Literature Review of Domain Adaptation with Unlabeled Data. Technical Report. University of Washington.

Mountney, P., Ionasec, R., Kaiser, M., Mamaghani, S., Wu, W., Chen, T., John, M., Boese, J., Comaniciu, D., 2012. Ultrasound and fluoroscopic images fusion by autonomous ultrasound probe detection, in: Proc MICCAI, Springer. pp. 544–551.

Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22, 1345–1359.

Shimodaira, H., 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. J Statistical Planning and Inference 90, 227–244.

Sugiyama, M., Kawanabe, M., Chui, P.L., 2010a. Dimensionality reduction for density ratio estimation in high-dimensional spaces. Neural Networks 23, 44–59.

Sugiyama, M., Suzuki, T., Kanamori, T., 2010b. Density ratio estimation: A comprehensive review, in: Proc Workshop on Statistical Experiment and Its Related Topics, Kyoto, Japan. pp. 10–31.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., Kawanabe, M., 2008. Direct importance estimation for covariate shift adaptation. Annals of the Institute of Statistical Mathematics 60, 699–746.

Tu, Z., 2005. Probabilistic boosting-tree: learning discriminative

models for classification, recognition, and clustering, in: Proc Int Conf on Computer Vision (ICCV), pp. 1589–1596.

Viola, P., Jones, M.L., 2004. Robust real-time face detection. International Journal of Computer Vision 57, 137–154.

Yamada, M., Sigal, L., Raptis, M., 2012. No bias left behind: Covariate shift adaptation for discriminative 3d pose estimation, in: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (Eds.), Proc European Conf on Computer Vision (ECCV), Springer Berlin Heidelberg. pp. 674–687.

Zheng, Y., Barbu, A., Georgescu, B., Scheuering, M., Comaniciu, D., 2008. Four-chamber heart modeling and automatic segmentation for 3D cardiac CT volumes using marginal space learning and steerable features. IEEE Trans Med Imaging 27, 1668–1681.

Zhu, X., 2008. Semi-Supervised Learning Literature Survey. Technical Report 1530. University of Wisconsin-Madison.