Localizing Target Structures in Ultrasound Video — A Phantom Study

R. Kwitt[1,*], N. Vasconcelos[b], S. Razzaque[c], S. Aylward[1]

[a] *Kitware Inc., Chapel Hill, NC, USA*
[b] *Statistical Visual Computing Laboratory, UC San Diego, USA*
[c] *InnerOptic Technology, NC, USA*

Abstract

The problem of localizing specific anatomic structures using ultrasound (US) video is considered. This involves automatically determining when an US probe is acquiring images of a previously defined object of interest, during the course of an US examination.

Localization using US is motivated by the increased availability of portable, low-cost US probes, which inspire applications where inexperienced personnel and even first-time users acquire US data that is then sent to experts for further assessment. This process is of particular interest for routine examinations in underserved populations as well as for patient triage after natural disasters and large-scale accidents, where experts may be in short supply.

The proposed localization approach is motivated by research in the area of dynamic texture analysis and leverages several recent advances in the field of activity recognition. For evaluation, we introduce an annotated and publicly available database of US video, acquired on three phantoms. Several experiments reveal the challenges of applying video analysis approaches to US images and demonstrate that good localization performance is possible with the proposed solution.

Keywords: Ultrasound imaging, Video analysis, Dynamic textures

1. Motivation

There are several reasons why 2-D, B-mode ultrasound (US) imaging is one of the prevalent imaging modalities in today's medicine. First, it is a highly versatile and non-invasive technology, applicable to first-care situations, routine examinations, and even for surgical guidance. Second, since the purchase, operation and maintenance costs of the equipment are moderate, there are various economic incentives for the adoption of US technology. Finally, US imaging has

*Corresponding author at: Kitware Inc., 101 East Weaver St, Carrboro, NC 27510, USA
Email address: roland.kwitt@kitware.com (R. Kwitt)

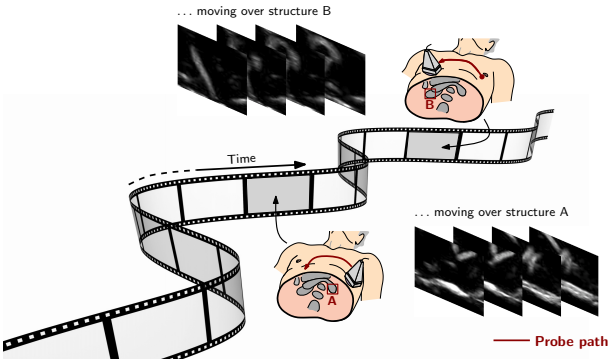


Figure 1: Localization of anatomical structures (e.g., structures A and B) by moving the Ultrasound probe along a path (dark red) on the human body.

become portable in the last few years, especially with the emergence of probes that can be connected to tablet PCs and cellphones. All of these properties make US a particularly attractive imaging modality for underserved areas of the world, where access to advanced tomographic imaging modalities, such as MRI or CT, is usually limited or impractical.

While there is little doubt about the diagnostic benefits of US in the hands of experienced clinicians, these benefits are less clear in situations where imaging must be performed by support staff or even inexperienced users. As with advanced tomographic imaging equipment, imaging experts are also a scarce resource in underserved areas of the world. Other scenarios in which expertise may be limited include extreme weather events, natural disasters, and other emergencies. Ideally, it would be possible to have non-experts acquire images, which would then be sent to experts for further analysis. It is therefore worth considering the development of technology that would allow non-experts to conduct US examinations. However, most research in medical vision has focused on the *assessment* component of the US examination pipeline, such as computer-aided diagnosis (i.e., predicting disease progression, tumor staging, etc.); however, very little work has been devoted to the very first step in the US examination pipeline, i.e., the *localization* of the area of the human body containing the anatomic features of interest. As illustrated in Fig. 1, the localization of these features involves the simultaneous 1) movement of a US probe across the body, and 2) observation of the resulting video stream, until the desired location is found. This task can be complex even for examiners with substantial experience and knowledge of the human anatomy. For novice users, the location of a certain anatomical structure (e.g., an organ) is often quite challenging. Hence, the development of automated methods for this localization would significantly enhance the ability of non-experts to operate US devices. Automated localization technology could also facilitate the *training* of radiology interns, by providing automated ground-truth for the localization task. This could be explored in the design of quantitative measures of learning progress, which could

be used to personalize the learning process. In fact, this technology could be useful even for experienced clinical personnel, since the localization of anatomical structures, such as vessels, tumors, etc., is usually the very first step of intraoperative ultrasound guidance. The most difficult part of this procedure is often the interpretation of the US images, namely how they relate to the physician's understanding of the anatomy and/or preoperative CT/MR studies. Automated localization procedures, especially if integrated into ultrasound scanners, could be used in the development of navigation systems that minimize these difficulties.

We are particularly interested in those anatomic imaging tasks where the target locations are pre-defined. For instance, the *FAST (Focussed Assessment of Sonography for Trauma (FAST)* [Scalea et al., 1999] procedure defines four locations on the human body that need to be assessed for the presence of free fluids. The FAST procedure is particularly important in cases of blunt abdominal trauma (common in car accidents, falls, violent crimes, and sporting accidents). The timely assessment of the FAST procedure can have drastic impact on patient treatment and outcome. Whether or not fluid is present then determines the path to take in a decision tree of triage and patient management.

The FAST procedure and other clinical tasks involving structure localization motivate the question of whether it is possible to exploit video analysis techniques to capture the discriminative information necessary for identifying what is imaged at any stage of an US examination. Although mapping a *single* US image to a particular anatomical location can be quite difficult, even for experienced physicians, access to the sequence of frames collected immediately before and after that image substantially simplifies the task. While the video dynamics may not be needed from a diagnostic point of view (i.e., for detecting disease or pathologies), they are an important cue for navigating the probe to a desired location. From this point of view, the problem of continuously searching for certain target structures in US video data is conceptually equivalent to the *event detection*, or *activity recognition* problems in computer vision. Nevertheless, a direct transfer of activity or event recognition methods to US imaging can be challenging, due to the differences in the acquisition of conventional versus US video (see Sect. 3). For other imaging modalities, such as CT or MRI, which produce large field-of-view volumetric images, the resulting data does not typically have a dynamic aspect or ambiguous localization. This could explain why recent advances in video analysis have received little attention in the medical imaging literature (see recent work in the recognition of surgical gestures from conventional video [Bejar et al., 2012], for an exception).

Contribution. The main contribution of this work is to propose an automated, template-based, approach to tackle the localization problem. This is defined as the ability to detect the presence (or absence) of certain target structures as the user acquires a continuous US video stream. We propose a *recognition by detection* paradigm, where a collection of short video *templates,* depicting the desired structures, are acquired at target locations (by expert users), off-line. These templates are continuously compared to the current video stream, using a similarity measure based on statistical models of the video. The proposed lo-

calization procedure exploits the fact that similarity is highest when the probe is moved over a target structure. The statistical models are based on the *kernel dynamic texture*, a recently introduced joint model of video appearance and dynamics. We extend this model with an enhanced observation component, based on the popular bag-of-words (BoW) representation from computer vision. For quantitative evaluation, we introduce a publicly-available and fully annotated database of US videos, acquired on three different phantoms. This facilitates assessment of the localization performance of the proposed system and establishes a baseline for future recognition experiments with US data. A number of experiments with this dataset demonstrate the feasibility of the proposed approach to US localization. A preliminary version of this work appeared in [Kwitt et al., 2012].

2. Related Work

The work presented in this article is conceptually related to various previous approaches to the problem of recognizing human activity in conventional video. There are, however, a number of significant differences. First, due to the strong emphasis on human activities, many of these approaches are specifically tailored to human behavior, e.g., relying on silhouette information [Blank et al., 2005], tracks [Kläser et al., 2010], or human pose information [Lv and Nevatia, 2007]) as important recognition clues. Second, activity recognition methods tend to depend on an effective temporal segmentation of the video into coherent parts, which depict a single activity at a time. However, because temporal segmentation can itself be quite challenging, the boundary between the two tasks is quite blurry. To account for this, recent works formulate recognition and temporal segmentation as a joint problem [Chen and Grauman, 2012; Hoai et al., 2011], or optimize the temporal segmentation according to the performance of an activity classifier [Satkin and Hebert, 2010]. In the context of continuous, real-time localization, it is impossible to assume either knowledge of the video clip boundaries, or availability of the full video.

The prevalent strategy for activity recognition is to build upon some mid-level representation of temporally segmented video clips. A video is represented by a collection of spatio-temporal descriptors, such as HOF/HOG [Laptev et al., 2008], HOG3D [Kläser et al., 2008], or extended SURF [Willems et al., 2008]. The descriptors are either computed at the nodes of a dense grid or at locations identified by interest point detectors (e.g., [Laptev and Lindeberg, 2003; Dollar et al., 2005; Willems et al., 2008]). Descriptor quantization with a sufficiently large codebook then produces the popular BoW representation, which can be fed to a discriminant classifier. Recently, representations at a higher, more abstract, level (e.g., action templates [Sadanand and Corso, 2012]) have achieved state-of-the-art results on various datasets.

An alternative strategy is to consider a video as a realization of a dynamical system. In this context, variants of the so called *dynamic texture* model [Doretto et al., 2003] have become increasingly popular for activity recognition. In particular, dynamic textures have been *kernelized* [Chan and Vasconcelos,

2007a], so as to capture a wide variety of motion patterns, used as components of mixture models [Chan and Vasconcelos, 2008], or even as *codebook* elements in a bag-of-dynamical-systems [Ravichandran et al., 2009; Coviello et al., 2012]. One characteristic that differentiates dynamic texture approaches from other works is that they capture both the appearance and the dynamics of the video, using a generative model. With the kernelized extension of Chan and Vasconcelos [2007a] it is even possible to capture the dynamics of features other than pixel values, e.g., histograms [Chaudhry et al., 2009], as long as a suitable kernel can be defined.

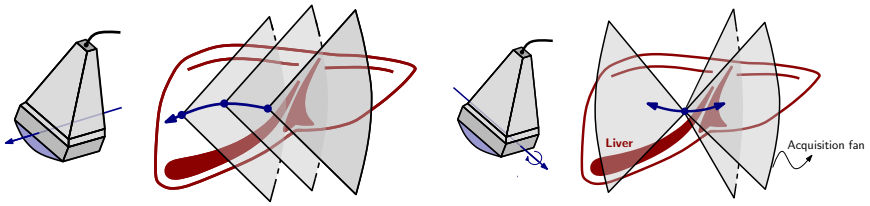
In this work, we explore the use of kernel dynamic textures (KDTs) for localization of anatomical features in US, using two different configurations. In the first, similar to [Chan and Vasconcelos, 2007a], we use KDTs to model the appearance and dynamics of raw B-mode intensity values in US video. In the second, we extend the approach proposed by Chaudhry et al. [2009] to model optical-flow histograms, by modeling US video with KDTs based on a mid-level BoW representation. The intuition is that augmenting the, already discriminative, BoW representation with a representation of feature dynamics should lead to a better, more robust, localization approach.

3. Ultrasound Video Acquisition

In this section, we briefly review the basics of the US acquisition process. This is mostly to facilitate the understanding of the difficulties involved in transferring, or extending, conventional video processing approaches to the domain of US. For a thorough introduction to US imaging we refer the reader to [Block, 2004].

From a physical point of view, the acquisition of US images can be summarized as follows. A US transducer emits short, directed, sound waves which are reflected and scattered by the underlying tissue layers. The reflected signal intensity is a function of the depth of the reflected structure. This intensity is displayed as a gray value. The prevalent form of US visualization is 2-D (realtime) *B-mode*, where a *slice-view* of underlying structures is acquired by composing single scanline measurements into an image. Depending on how the piezoelectric elements are arranged in the US probe (e.g., a linear or curved array), the resulting images are either rectangular or fan-like.

Due to the slice-view nature of US imaging, there is no clear notion of an object in a single US frame. This is in contrast to conventional video cameras, which capture *snapshots* of the surrounding environment at a specific frame rate. In fact, US only allows the observation of a very limited portion of any actual *physical* object. This induces large variations in visual appearance, depending on the angle of the imaging plane, and obviously confounds the adoption of frame-based object detectors developed for conventional images. Two US transducer movements, illustrated in Fig. 2 for a schematic liver model, are particularly common in clinical practice: *translation* and *tilting*. Translation shifts the imaging plane, while titling leads to fan-like acquisitions. An anatomical structure, such as an organ, visible in one frame can suddenly disappear, if



(a) Translation **(b)** Tilting

Figure 2: Illustration of the acquisition fan — on a schematic liver model (dark red) — when moving the US probe according to two common probe movements: translation and tilting.

the imaging plane no longer intersects the target object. This changes the appearance of image patches over time and precludes the assumption of *brightness constancy*. Since this is a core assumption in optical-flow estimation [Sun et al., 2010], approaches that rely on the measurement of displacement vectors (e.g., [Chaudhry et al., 2009]) are ill-suited for US video.

Another characteristic of US data are image acquisition *artifacts*, most notably *speckle noise* and *acoustic shadowing*. Speckle noise results from the interference of sound waves and is common in any coherent imaging modality. Acoustic shadowing occurs when the signal is either totally reflected (e.g., at air) or absorbed (e.g., at bone material). This results in either completely black or white regions in the US image.

4. Localization in Ultrasound Video

The proposed localization procedure has two components. The first is a template acquisition step, performed offline, by an US examination expert. This consists of acquiring a database of short video sequences, referred to as *templates*, by moving the probe over a set of target regions on the human body. These target regions are chosen so as to ensure that a set of predefined underlying structures are clearly visible. They could be the four regions of the FAST procedure (see Sect. 1) for instance. Multiple templates could be imaged per anatomic structure, using multiple patients and probe orientation/angle settings. The second component, which is executed *during* the actual US examination, follows a *recognition by detection* paradigm. It consists of comparing, in real-time, a sliding-window of the current video stream to all templates. As new frames are acquired, the sliding window is moved forward. By defining a similarity measure between two video sequences, or corresponding statistical models, it is possible to evaluate whether the video in the sliding window resembles one of the database templates. This facilitates to provide immediate feedback to the user and specifically enables localization *during* the procedure. Since, there can be sequences which resemble none of the templates, there is a need to consider a *Null* class.

To model US video, we resort to statistical models of joint video appearance and dynamics from the computer vision literature. In particular, we leverage

the non-linear extension of the dynamic texture model of [Doretto et al., 2003] known as the kernel dynamic texture (KDT) [Chan and Vasconcelos, 2007b]. KDTs offer several advantages over dynamic textures as a model of video dynamics, including the ability to capture non-linear dynamics and the ability to rely on observations other than intensity values. There are also a variety of KDT similarity measures, including the (kernelized) variant of the Martin distance of [Martin, 2000; Chan and Vasconcelos, 2007b], or the Binet-Cauchy kernel of Vishwanathan et al. [2007]. In what follows, we will discuss two configurations of KDTs. The first relies on the raw intensity values of B-mode US as observations. The second uses BoW histograms. The latter configuration is denoted as *Bag-of-Word Dynamics (BoWDyn)*. We also show how to use the similarity measurements in a kernel density estimation framework to naturally facilitate inclusion of the *Null* class.

4.1. Notation

Unless otherwise stated, we adhere to the following notational conventions. Vectors are always column vectors, denoted by lower case boldface letters (e.g., $\boldsymbol{x}$). $\boldsymbol{x}^\top$ denotes the vector transpose. Matrices are denoted by upper case boldface letters (e.g., $\boldsymbol{X}$). By X_{ij}, we refer to the (i,j)-th entry of matrix $\boldsymbol{X}$. The notation $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N], \boldsymbol{x}_i \in \mathbb{R}^d$, denotes that matrix $\boldsymbol{X}$ is composed of N d-dimensional vectors, hence $\boldsymbol{X}$ is a $d \times N$ matrix. Lower-case letters (e.g., γ) denote scalars, upper-case letters (e.g., Y) usually denote random variables. Further, $\boldsymbol{I}$ denotes the identity matrix and $\boldsymbol{e}$ denotes a vector of all ones.

4.2. Kernel Dynamic Textures

Since dynamic texture models and their kernel extension form the core of our approach, we briefly review these models and the associated parameter estimation procedures. For further details, we refer to the original works of Doretto et al. [2003] and Chan and Vasconcelos [2007a,b].

In the following, an US video is considered as an ordered sequence of T video frames, arranged in an observation matrix $\boldsymbol{Y} = [\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{T-1}]$, where $\boldsymbol{y}_t \in \mathbb{R}^d$ is the frame observed at time t, represented by its intensity values. Under the DT framework of Doretto et al. [2003], these observations are modeled as samples of a *linear dynamical system (LDS)*. At time t, a vector of state coefficients $\boldsymbol{x}_t \in \mathbb{R}^T$ is first sampled from a first-order Gauss-Markov process, and the state coefficients are then linearly combined into the observed video frame $\boldsymbol{y}_t$, according to

$$\boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{w}_t, \tag{1}$$

$$\boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}_t, \tag{2}$$

where $\boldsymbol{A} \in \mathbb{R}^{T \times T}$ is the *state-transition* matrix and $\boldsymbol{C} \in \mathbb{R}^{d \times T}$ is the *generative* matrix that governs how the state determines the observation. Further, $\boldsymbol{w}_t \in \mathbb{R}^T$ and $\boldsymbol{v}_t \in \mathbb{R}^d$ denote state and observation noise with $\boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and $\boldsymbol{v}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$, respectively.

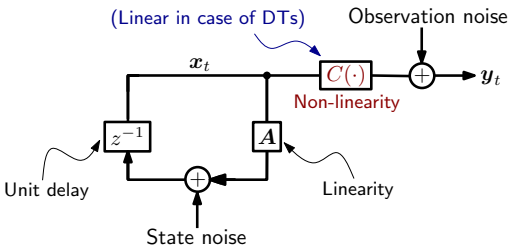


Figure 3: Generative model for DTs and KDTs.

Assuming that the observations are centered (which is straightforward by subtracting the column-wise means of $\boldsymbol{Y}$) and following the system identification strategy of Doretto et al. [2003], $\boldsymbol{C}$ can be estimated by computing an SVD decomposition of the observation matrix $\boldsymbol{Y}$ as $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ and setting $\boldsymbol{C} = \boldsymbol{U}$. The state matrix $\boldsymbol{X} = [\boldsymbol{x}_0 \cdots \boldsymbol{x}_{T-1}]$ is estimated as $\boldsymbol{X} = \boldsymbol{\Sigma}\boldsymbol{V}^\top$ and $\boldsymbol{A}$ can be computed using least-squares as $\boldsymbol{A} = [\boldsymbol{x}_1 \cdots \boldsymbol{x}_{T-1}][\boldsymbol{x}_0 \cdots \boldsymbol{x}_{T-2}]^\dagger$, where † denotes the pseudoinverse. When restricting the DT model to N states, $\boldsymbol{C}$ is restricted to the N eigenvectors corresponding to the N largest eigenvalues, computed in the SVD decomposition of $\boldsymbol{Y}$. The rest follows accordingly.

In the non-linear DT extension of Chan and Vasconcelos [2007b], the generative matrix $\boldsymbol{C}$ is replaced by a non-linear observation function $C : \mathbb{R}^T \to \mathbb{R}^d$, i.e.,

$$\boldsymbol{y}_t = C(\boldsymbol{x}_t) + \boldsymbol{v}_t, \tag{3}$$

while the state equation remains linear, cf. (1). The corresponding dynamical system, denoted a *kernel dynamic texture (KDT)*, is illustrated in Fig. 3. Due to the non-linearity of C, the KDT requires a different, although conceptually equivalent, set of parameter estimates. The idea is to learn the inverse mapping $D : \mathbb{R}^d \to \mathbb{R}^T$ from observation to state space, using a kernel principal component analysis (KPCA). The KPCA coefficients then represent the state variables. Given the kernel matrix $K_{ij} = k(\boldsymbol{y}_i, \boldsymbol{y}_j)$ arising from a suitable[1] kernel function k, the computation of KPCA coefficients is performed in three steps. First, the kernel matrix is centered in feature space, i.e., $\tilde{\boldsymbol{K}} = (\boldsymbol{I} - {}^1\!/_N \boldsymbol{e}\boldsymbol{e}^\top)\boldsymbol{K}(\boldsymbol{I} - {}^1\!/_N \boldsymbol{e}\boldsymbol{e}^\top)$[2]. The eigenvector/eigenvalue pairs $(\lambda_i, \boldsymbol{v}_i)$ of $\tilde{\boldsymbol{K}}$ are then determined. Finally, the KPCA coefficients $\boldsymbol{X}$ are computed as $\boldsymbol{X} = \boldsymbol{\alpha}^\top \tilde{\boldsymbol{K}}$, where $\boldsymbol{\alpha}$ is the $T \times N$ KPCA weight vector matrix $[\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N]$ with $\boldsymbol{\alpha}_i = \lambda_i^{-1/2} \boldsymbol{v}_i$.

The ability of the KDT to capture the appearance and dynamics of US sequences is illustrated in Fig. 4. The middle row of the figure presents a clinical US video, acquired by sweeping a US probe on the surface of a patient liver (shown in top row), during a radio-frequency tumor ablation. The bottom row shows a synthetic sequence sampled from the KDT model of 10 states (using a

[1] (conditionally) positive definite
[2] $\boldsymbol{e}$ is a N-dimensional column vector of all ones.

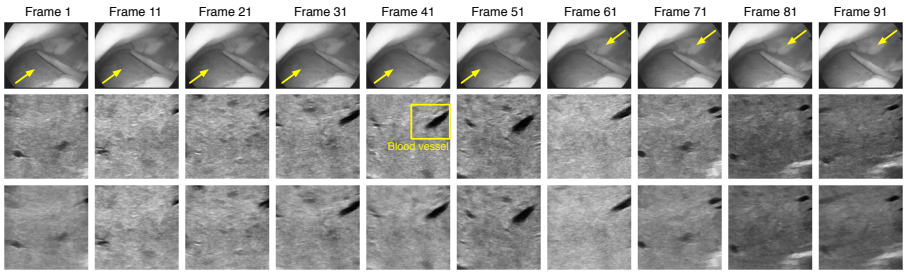


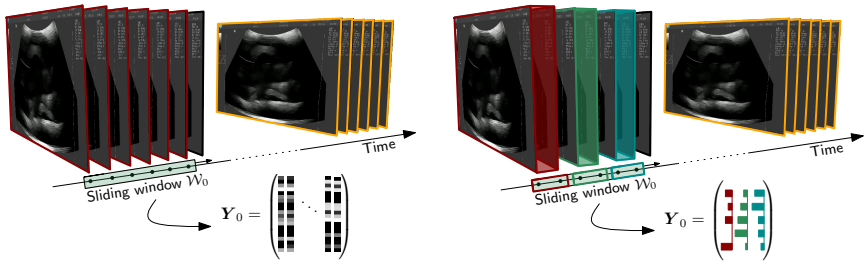
Figure 4: Clinical data from one patient, acquired during a radio-frequency ablation of a liver tumor. The probe sweeps from left to right and back (see arrows) on the liver surface. The *top* row shows a video of the probe movement; the *middle* row shows a subset of the actual US frames with a blood vessel moving in and out of the imaging plane (highlighted); the *bottom* row shows the same set of frames, *synthesized* from a KDT model with 10 states (using a RBF kernel) that was estimated from the original US sequence.

RBF kernel) that best fits the US sequence. The quality of the reconstruction supports the claim that the KDT can capture the dynamics of clinical US material. However, recent work in computer vision has shown that a frame-based representation is not necessarily the most robust to noise, appearance changes due to pose variability, variable imaging conditions, etc. In domains like US, where these types of nuisances are prevalent, it is advisable to adopt histogram-based representations, which usually exhibit significantly greater robustness to imaging variability. An important point, in this context, is that KDTs are not restricted to intensity observation matrices, but can rely on any kind of features for which it is possible to define a kernel (e.g., RBF).

4.3. Bag-of-Word Dynamics

In the dynamic texture recognition literature, DTs and KDTs were originally proposed to model the appearance and dynamics of video frames (cf. [Doretto et al., 2003; Chan and Vasconcelos, 2007b; Vishwanathan et al., 2007]). Recently, Chaudhry et al. [2009] have shown that the application of KDTs to histograms of oriented optical flow (HOOF) achieve state-of-the-art results for human activity recognition. While, as discussed in Sect. 4, HOOF features are not applicable to US, the use KDTs as models of the dynamics of mid-level features has properties of interest for US localization. In addition to greater robustness to imaging variability, these representations enable reduced model estimation complexity, since mid-level features are usually lower-dimensional than video frames. This leads to smaller observation matrices $\boldsymbol{Y}$, allowing a faster computation of KPCA. We will later see that the similarity computation also benefits from this reduced dimensionality, which also leads to a smaller memory footprint for the whole model. This is particularly interesting in the context of mobile devices.

To leverage recent advances in mid-level video representation (cf. Sect. 2), a video of T frames is first divided into G overlapping frame groups, using a sliding window denoted as the *grouping window*. Each group of frames is then



(a) Pixel-based **(b)** Mid-level representation based

Figure 5: Composition of the observation matrix $\boldsymbol{Y}$ — for the first sliding window $\mathcal{W}_0$ — either based on a raw pixel-intensities or b mid-level feature histograms computed from sub-sequences within $\mathcal{W}_0$ (in this example, the subsequences do not overlap).

represented as a BoW histogram, built upon some low-level spatio-temporal descriptor. As illustrated in Fig. 5b, each column of the observation matrix $\boldsymbol{Y}$ contains a normalized BoW histogram, instead of a vector of intensity values. The KDT model, estimated from $\boldsymbol{Y}$, captures the dynamics of the *histogram change* from frame group to frame group. This is more discriminative than a frame-based BoW model, since it accounts for video dynamics. When compared to standard dynamic textures, the use of the mid-level representation guarantees improved robustness to appearance variability.

In this work, the descriptors are based on HOG3D [Kläser et al., 2008] and sampled on an evenly spaced spatio-temporal grid. We do not rely on interest point detectors, since their computation in US is highly susceptible to speckle noise. While the use of HOG3D descriptors is somewhat arbitrary, we emphasize that the concept of modeling the dynamics of a mid-level representation is generic and not tied to a specific low-level descriptor.

4.4. Sliding-Window Localization

In this section, we discuss the proposed localization strategy. A database of M US video templates $\mathcal{T}_0, \ldots, \mathcal{T}_{M-1}$, acquired from C different anatomical structures, is first assembled. Usually this includes multiple templates per structure, $C \ll M$, e.g., acquired from different directions. A KDT is then learned for each template. Let $m = \text{median}(|\mathcal{T}_i|)$ be the median length (in frames) of the template videos. A temporal sliding window $\mathcal{W}_t$, containing the last m frames of the current video stream, is used to estimate a KDT. This KDT is then compared to the M template KDTs in the database[3]. The window is finally shifted forward by s frames, as s new frames are acquired by the US system, and the process repeated.

Assuming a suitable similarity measure between KDTs, this produces M similarity measurements $d_t^0, \ldots, d_t^{M-1}$ at location t. In what follows we assume

[3]If the goal is to localize one particular structure, the comparison can be restricted to the subset of template models corresponding to that particular structure.

that lower values of d_t^j indicate a higher degree of similarity. To account for the fact that no target structure may be present at the location (i.e., the *Null* class), an indicator function $i : \mathbb{N}_0 \to \{1, \ldots, C\} \cup \{\text{Null}\}$ is defined as follows

$$i(t) := \begin{cases} c(p), & \text{if } d_t^p < \gamma \\ \text{Null}, & \text{else.} \end{cases} \qquad \text{with } p := \arg\min_j d_t^j \ . \tag{4}$$

The function $c : \{0, \ldots, M-1\} \to \{1, \ldots, C\}$ is a label function that maps a template index p to one of the C target structures. Depending on the threshold γ, $i(t)$ indicates the presence, or absence, of one of the target structures. The assumption is that templates containing the structure that is currently observed will not only lead to the minimum distance measurement but will also have a characteristically low value.

While many activity recognition methods employ a support vector machine (trained on positive and negative instances) as the final recognition stage (cf. Duchenne et al. [2009]; Willems et al. [2009]; Kläser et al. [2010]), we propose a non-parametric strategy for threshold determination. This enables better control of the false-positive rate. Let H_0 denote the null-hypothesis of *no matching template* (i.e., the Null class). The goal is to control $\alpha = \mathbb{P}(d_t^p < \gamma | H_0)$, i.e., the probability of the indicator function being non-Null when H_0 holds. Given α, the determination of γ requires an estimate of the probability density of d_t^p under H_0, i.e., $p(d_t^p | H_0)$. This can be accomplished with a Parzen window estimate

$$\hat{p}(d|H_0) = \frac{1}{S} \sum_{s=0}^{S-1} w(d - d_s, h), \tag{5}$$

based on a collection $\{d_s\}_{s=1}^S$ of measurements under H_0. These are similarity values collected from US videos that do not contain any instance of a template in the database. The function $w(d, h)$ is a kernel (e.g., Gaussian) of width h. Since d is one-dimensional, the tuning of the kernel width h is not difficult[4]. Given $\hat{p}(d|H_0)$, γ is computed with recourse to the inverse of the corresponding cumulative distribution function, i.e., $\gamma = F^{-1}(\alpha)$ (solved numerically). An interesting property of this threshold computation strategy is that it only requires negative training data. While negative samples are relatively easy to acquire, even by novice users, the assembly of positive examples (matches to the templates) requires more knowledge and experience in the US imaging process.

4.5. Measuring model-to-model similarity

Several measures of similarity between linear dynamical systems have been proposed in the literature. The Martin distance [Martin, 2000], the Binet-Cauchy kernel of Vishwanathan et al. [2007], and information-theoretic measures such as the KL-divergence of Chan and Vasconcelos [2005] are popular choices.

[4] In practice, even the *normal distribution approximation* $h = 1.06\hat{\sigma}S^{-1/5}$ works well, where σ is an (robust) estimator for the spread of the data.

Recently, both the Martin distance and the Binet-Cauchy kernel have been extended to KDTs [Chan and Vasconcelos, 2007b; Chaudhry et al., 2009]. In this work, we adopt the Martin distance. Given two videos, represented by their DT models, $\mathcal{M}_a = (\boldsymbol{A}_a, \boldsymbol{C}_a)$ and $\mathcal{M}_b = (\boldsymbol{A}_b, \boldsymbol{C}_b)$ (of N states each), this distance is defined as [Martin, 2000; De Cock and Moore, 2000]

$$d^2(\mathcal{M}_a, \mathcal{M}_b) = -\log \prod_{i=1}^{N} \cos^2(\phi_i), \tag{6}$$

where ϕ_i are subspace angles between the infinite observability matrices $\boldsymbol{O}_a = [\boldsymbol{C}_a^\top \ (\boldsymbol{C}_a\boldsymbol{A}_a)^\top \ (\boldsymbol{C}_a\boldsymbol{A}_a^2)^\top \cdots]^\top$ and $\boldsymbol{O}_b = [\boldsymbol{C}_b^\top \ (\boldsymbol{C}_b\boldsymbol{A}_b)^\top \ (\boldsymbol{C}_b\boldsymbol{A}_b^2)^\top \cdots]^\top$. It can be shown that the $\cos(\phi_i)$ terms are the N largest eigenvalues λ_i of the generalized eigenvalue problem

$$\begin{bmatrix} \boldsymbol{0} & \boldsymbol{O}_{ab} \\ \boldsymbol{O}_{ba} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \lambda \begin{bmatrix} \boldsymbol{O}_{aa} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{O}_{bb} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} \tag{7}$$

with

$$\boldsymbol{O}_{ab} = \boldsymbol{O}_a^\top \boldsymbol{O}_b \tag{8}$$

subject to $\boldsymbol{x}^\top \boldsymbol{O}_{aa} \boldsymbol{x} = 1$ and $\boldsymbol{y}^\top \boldsymbol{O}_{bb} \boldsymbol{y} = 1$. For DTs, computation of $\boldsymbol{O}_{ab}$ is straightforward, since

$$\boldsymbol{O}_{ab} = \sum_{n=0}^{\infty} (\boldsymbol{A}_a^n)^\top \boldsymbol{C}_a^\top \boldsymbol{C}_b \boldsymbol{A}_b^n \tag{9}$$

and the terms $\boldsymbol{C}_a^\top \boldsymbol{C}_b$ can be evaluated. For KDTs, it can be shown that the computation of $\boldsymbol{C}_a^\top \boldsymbol{C}_b$ reduces to computing the inner products between the principal components of kernel matrices $K_{ij}^a = k(\boldsymbol{y}_i^a, \boldsymbol{y}_j^a)$ and $K_{ij}^b = k(\boldsymbol{y}_i^b, \boldsymbol{y}_j^b)$, i.e.,

$$\boldsymbol{O}_{ab} = \sum_{n=0}^{\infty} (\boldsymbol{A}_a^n)^\top \tilde{\boldsymbol{\alpha}}^\top \boldsymbol{G} \tilde{\boldsymbol{\beta}} \boldsymbol{A}_b^n, \tag{10}$$

where $\tilde{\boldsymbol{\alpha}} = [\tilde{\boldsymbol{\alpha}}_0 \cdots \tilde{\boldsymbol{\alpha}}_{T-1}]$, $\tilde{\boldsymbol{\beta}} = [\tilde{\boldsymbol{\beta}}_0 \cdots \tilde{\boldsymbol{\beta}}_{T-1}]$ are the (normalized) KPCA weight matrices with $\tilde{\boldsymbol{\alpha}}_i = \boldsymbol{\alpha}_i - {}^1\!/_N(\boldsymbol{e}^\top \boldsymbol{\alpha}_i)\boldsymbol{e}$, $\tilde{\boldsymbol{\beta}}_i = \boldsymbol{\beta}_i - {}^1\!/_N(\boldsymbol{e}^\top \boldsymbol{\beta}_i)\boldsymbol{e}$ and $\boldsymbol{G}$ is the kernel matrix with entries $G_{ij} = k(\boldsymbol{y}_i^a, \boldsymbol{y}_j^b)$. In the remainder of this work, we use (6) – (10) to measure similarity between US sequences, modeled as KTDs, and a standard RBF kernel for all kernel computations[5]. It should be noted that, since the computation of $\boldsymbol{G}$ requires the observations $\boldsymbol{y}_i^a$ and $\boldsymbol{y}_j^b$, these must be stored along with the template KDT parameters. Hence, replacing raw intensities with the mid-level BoW representation of groups of frames can considerably reduce the memory footprint of each model.

[5]For KPCA, the kernel width is set to $\sigma^2 = \text{median}_{i,j}\|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2$; to compute G_{ij}, it can be shown [Chan and Vasconcelos, 2007b] that $\boldsymbol{y}_i^a$ and $\boldsymbol{y}_j^b$ must be scaled by σ_a and σ_b.

Figure 6: Schematic illustration of the US phantom (left); three images of one of our test phantoms at different viewing angles (right).

5. Experiments

The localization performance of the proposed approach was evaluated on phantom data. However, conventional phantoms have significant limitations for recognition. This follows from the fact that medical imaging phantoms are designed for applications, such as surgical training and evaluation of registration or segmentation algorithms, with limited variability in visual appearance and relatively small amounts of noise and artifacts. This is particularly true for small-scale phantoms. For these reasons, we decided to prepare three *custom* phantoms, made of gelatin and Soba noodles. These allow some control over the variety of the underlying structures and have been previously shown useful for US image analysis [Aylward et al., 2002].

To create these *localization phantoms*, we first loosely attached a set of noodles to a grid of thin strings, laid out across a large bowl. We then filled the bowl with hot gelatin and let it cool. As the gelatin changed state from liquid to solid, it formed the embedding mass. The resulting phantoms have a number of interesting properties for recognition: noodles are self-similar at a small scale, have ambiguous patterns of bends at medium and large scales and, in general, present a rich set of structures that are difficult distinguish through casual inspection. A schematic illustration of a typical gelatin phantom is shown in Fig. 6, together with a set of images of actual phantoms, taken from various viewing angles. In all experiments, phantoms are mounted on a plastic plate, causing reflections due to the reflectance of sound waves on the plastic surface. Further artifacts result from small air pockets within the gelatin, causing the signal to scatter. All these artifacts lead to realistic US data.

The US database consists of videos from three different phantoms, denoted as phantoms A, B and C. While the three contain roughly the same number of noodles, these vary in length, radius and structure. Imaging was based on the Telemed LogicScan 128 INT-1Z kit, which allows video capture in uncompressed AVI format. The US frequency was set to 5Mhz, and the penetration depth to 90mm. *Speckle reduction* was enabled in the US acquisition software and all videos were acquired freehand, without tracking.

5.1. Acquisition protocol

Nine different noodle structures were first identified, including *straight noodle segments*, *crossing noodles,* and *loops*. The gelatin medium turned out to be

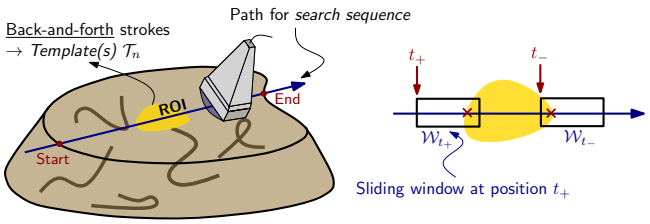


Figure 7: Acquisition of the search path and template sequences (left). Definition of the earliest and latest localization times t_+, t_- (right).

particularly convenient for the purpose of locating interesting targets, since its transparency allows for visual inspection of the underlying structures.

Search path video acquisition. For each target structure, one *search path* video was recorded as illustrated in Fig. 7. Video lengths range from 188 to 322 frames, with a medium length of 249 frames. In absolute time, this translates into 6.4 to 10.7 seconds. Probe movement was restricted to translation. While six out of nine search path sequences were acquired along a roughly linear path (from one end of the phantom to the other, cf. Fig. 7) and only contain a target once, the remaining three show two occurrences of the target (acquired by moving the probe along a circle, visiting the target twice). We adhered to the guideline of keeping the probe as perpendicular as possible to the phantom's surface. While this not a strict requirement, it ensures that the US fan sufficiently penetrates the underlying tissue.

Template video acquisition. To acquire the *template* videos to be stored in the database, the nine targets were first visually located on the phantoms. Short strokes were then performed at the identified locations. The probe orientation was roughly similar to the probe orientation used to acquire the search path videos. It should be noted, however, that the templates were not acquired immediately after the search paths. To avoid biasing the experiments, templates and search paths were collected at different acquisition times, typically on different days. All templates were (temporally) clipped to the median template video length of m frames. Since US video usually contains meta information about the device settings, we further clipped each video (both templates and search paths) spatially, to a 128×128 pixel region. To minimize biases due to this cropping, we randomly displaced the cropping window by d pixels in each direction and added the clipped videos to the template database. In all experiments, we used $d = 10$ and sampled 10 positions per template, leading to a database of 90 templates in total.

5.2. Ground truth annotation

To create ground truth annotations, i.e., marking the appearance and disappearance of a target structure, we used VATIC [Vondrick et al., 2012]. This allows a user to quickly draw and modify bounding boxes, on a keyframe basis, around structures of interest. While the actual bounding boxes are of less

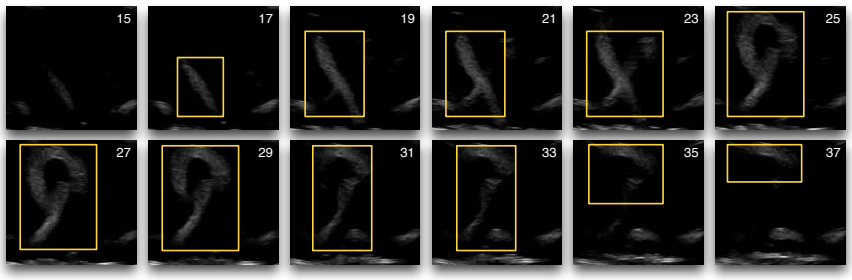


Figure 8: Annotation example for a search path; bounding boxes around the structure of interest, i.e., a loop, (positioned using VATIC [Vondrick et al., 2012]) are shown in yellow. Frame numbers 17 and 37 correspond to positions t_+ and t_- (best viewed in color).

interest in our context, the placement of the first and last bounding box are particularly important, since they mark the points where a target structure moves in and out of the imaging plane. Since video annotation is an inherently subjective task, we provided our template database to five different persons and asked them to review the corresponding search sequences (one at a time). Comparing the search path video with the templates was allowed, to simplify the task and simulate the situation of an expert ground truth annotator. We also allowed the annotator to move back and forth in the search path video and modify, or remove, bounding boxes. Each annotator was given the objective of placing a bounding box around the area where a target structure moves into the imaging plane; then adjust the bounding box as long as the structure is present or changes its appearance; and finally remove the bounding box once the structure is no longer visible. Figure 8 presents an example of an annotated search path. As shown in Fig. 7, the frame numbers for the *first* and *last* bounding box make up the tuple (t_+, t_-). The ground-truth localization interval of each search path sequence was defined as the component-wise median over the five tuples (one per annotator).

5.3. Evaluation Metrics

The evaluation metric for the localization task requires careful consideration, since different metrics highlight different characteristics [Ward et al., 2011]. Although performance was evaluated on a frame-by-frame basis, it should be noted that the finest possible granularity for localization is the sliding window shift.

Under the threshold determination strategy of Sect. 4.4, it is possible to set a desired false-alarm rate α and define a simple event-based performance criterion. Given the minimum distance d_t^p at position t (associated with the p-th database template), true and false positive rates (TP,FP) are defined for search path j according to

$$\text{TP} = (i(t) = j) \wedge (t_+ < t < t_-) \tag{11}$$

$$\text{FP} = (d_t^p < \gamma) \text{ and } t \notin [t_+, t_-]. \tag{12}$$

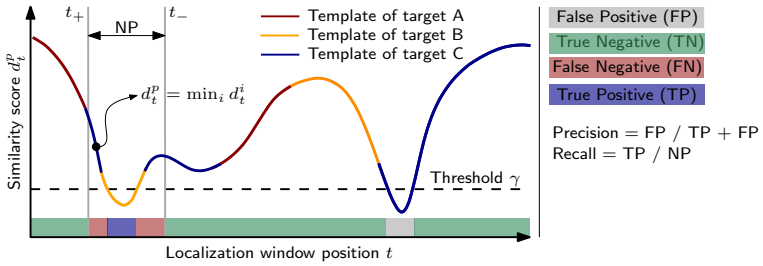


Figure 9: Definition of false positive/negatives and true positives/negatives in the context of the localization problem, illustrated on a toy example with three target structures. The similarity curve shows the similarity measure d_t^p (cf. Sect. 4.4) for a search path containing structure B.

Whether $c(p) = j$ does not make a difference for a false positive. Fig. 9 illustrates the definition of the performance measures on an example similarity curve, for a fictional search path containing structure B. To visualize performance, it is customary to vary α and obtain precision/recall (PR) curves. We also compute the *average precision (AP)* per search path. As a summary statistic, we adopt the mean AP (mAP), taken over all search path AP values.

5.4. Implementation

The proposed localization method was implemented with a combination of C++ and MATLAB. In this implementation, KDTs are estimated using the KPCA-based strategy of Chan and Vasconcelos [2007a], briefly outlined in Sect. 4.2. As low-level features for BoWDyn, we use dense HOG3D descriptors [Kläser et al., 2008][6] and standard K-Means clustering for codebook generation (initialized using K-Means++). All experiments used a Gaussian RBF kernel, with kernel-width σ set to the column-wise median of the observation matrix. Further configuration details are given in the discussion of the corresponding experiments.

5.5. Results

We started by evaluating the performance of 1) KDTs estimated from intensity values (IntDyn) and 2) KDTs estimated from BoW histograms (BoWDyn). The two methods were compared to a commonly-used SVM classifier. In all cases, the localization window length was set to the median template length of $m = 40$ frames, and the sliding window shift to $s = 2$ frames, so as to ensure a smooth change in the localization measure. For BoWDyn, the internal grouping window size was set to ten frames, with a shift of two frames.

Tables 1a and 1b list the mAP value for each configuration of IntDyn and BoWDyn, the minimum and maximum AP value, and the Precision/Recall (PR) values at the threshold γ determined with $\alpha = 0.05$. For both IntDyn and

[6] Online available from http://lear.inrialpes.fr/people/klaeser/research_hog3d

States	mAP	min/max AP	PR, $\alpha = 0.05$
2	50.6	2.8 / 94.0	50.5 / 55.7
5	58.7	0.0 / 96.9	62.9 / 58.2
8	55.5	1.4 / 95.6	58.4 / 57.8

(a) IntDyn

States	#Codewords	mAP	min/max AP	P/R, $\alpha = 0.05$
2	50	48.8	16.3 / 88.5	63.1 / 48.6
	100	59.3	19.1 / 94.2	67.6 / 57.6
	400	58.9	45.4 / 80.7	74.5 / 59.2
5	50	52.3	9.4 / 88.8	65.2 / 54.0
	100	51.6	15.1 / 88.9	66.0 / 66.4
	400	61.1	39.0 / 87.6	70.1 / 58.5
8	50	46.9	5.1 / 73.8	63.5 / 49.5
	100	50.4	26.5 / 96.7	64.4 / 54.6
	400	63.4	45.9 / 91.3	70.8 / 53.4

(b) BoWDyn

Table 1: Localization results for a IntDyn and b BoWDyn with a varying number of states and codebook size on 64×64 videos.

BoWDyn, the number of KDT states was varied from 2 to 8. For BoWDyn, the codebook size ranged from 50 to 400. A comparison of the mAPs achieved with IntDyn and BoWDyn shows that the latter can achieve equal, or better, localization performance even at moderate codebook sizes (100 codewords). In fact, a KDT with 2 states and a 100-dimensional BoW histogram achieves better mAP performance (59.3%) than the best result of Table 1a (58.7%). This is appealing for several reasons. First, a 100-dimensional BoW histogram only requires a fraction of the storage required per US frame. Second, a BoW histogram is only required per frame grouping (15 histograms per template in these experiments), while each frame must be stored for IntDyn. Hence, the latter has a much larger memory footprint.

Regarding the sensitivity to the number of KDT states, Tables 1a and 1b suggest that performance usually improves with the number of states. However, this trend is not evident for all configurations. For IntDyn, an increase in the number of states from 5 to 8 actually leads to a decrease in performance. This could be explained by the fact that 5 states may be sufficient to capture the appearance variability of the US videos. In fact, when KPCA is used, only a very small portion of the variation in the kernel-induced feature space is explained by the additional three eigenvalues. This leads to a less stable estimate of the KDT parameters, rendering the similarity measurement more difficult. For BoWDyn, a higher number of states appears to be beneficial only for larger codebooks. For example, with a codebook size of 400, additional states lead to a localization performance increase from 58.9% to 63.4%, the highest mAP value of this experiment. A comparison of the PR curves (per search path) for the

best IntDyn and BoWDyn configuration is shown in Fig. 10. The figures reveal that none of the two approaches clearly outperforms the other. Given this, the advantages of BoWDyn in terms of storage make it a superior representation.

While providing a good summary of localization performance, mAP values can be sometimes misleading. For example, by inspecting the PR values at $\alpha = 0.05$, it can be seen that precision tends to be higher than recall. While high precision implies that whatever is identified as target is very likely to be a target, recall summarizes the percentage of targets identified by the human annotator that are actually detected. Low recall values are somewhat inevitable, since the human ground truth localization interval $[t_+, t_-]$ tends to be large. This simply means that, as shown in Fig. 8, the human annotators are particularly good at identifying target structures once they start to appear in the imaging plane, and track them until they completely disappear. In contrast, the proposed localization methods typically detect targets once the localization window substantially overlaps the structure. Our evaluation metric, however, is designed to consider a measurement $d_t^p > \gamma, c(p) = j$ as a false negative on search path j as long as the localization window overlaps the ground truth interval $[t_+, t_-]$. On the one hand, we consider this as a reasonable choice since, in principle, it is possible to detect the structure at all those positions. On the other hand, this evaluation setup has the effect that the overall performance appears mediocre when compared to the human annotations. In practice, though, the only concern is *if* we can recognize a structure or not. While this objective is better captured by a solely *event-based* evaluation metric[7], our limited sample size would only allow for a very coarse (since we only have nine events) comparison between approaches. Nevertheless, performance improvements can be achieved in multiple ways. One strategy could be to shrink the length of the localization window to an optimal choice. This essentially means balancing the gains in reduced detection delay against potential losses in robustness due to reduced information within one window. In general, however, system parameters will eventually depend on the size of the target structure(s).

Comparison to the State-of-the-Art. In Sect. 3 we noted that optical-flow based approaches are ill-suited for US video. An experiment was designed to confirm this, by learning KDTs over the HOOF features of Chaudhry et al. [2009]. The number of HOOF bins was varied from 8 to 100 and optical flow was computed with both the original formulation of Horn and Schnuck [Horn and Schnuck, 1981], and the *Classic+NL* method of [Sun et al., 2010]. As expected, the performance was poor. We could not find a combination of optical-flow technique and bin size capable of reaching mAP values beyond 9%. This is worse than random choice, for all search paths.

Finally, we evaluated the popular strategy, for activity recognition, of using the BoW mid-level representation as feature space for an SVM classifier. The configuration is similar to BoWDyn. Dense HOG3D descriptors were extracted from 1) the templates and 2) a random sample of *Null* class sliding windows

[7] BoWDyn, for instance, allows localization in 8/9 cases at $\alpha = 0.05$.

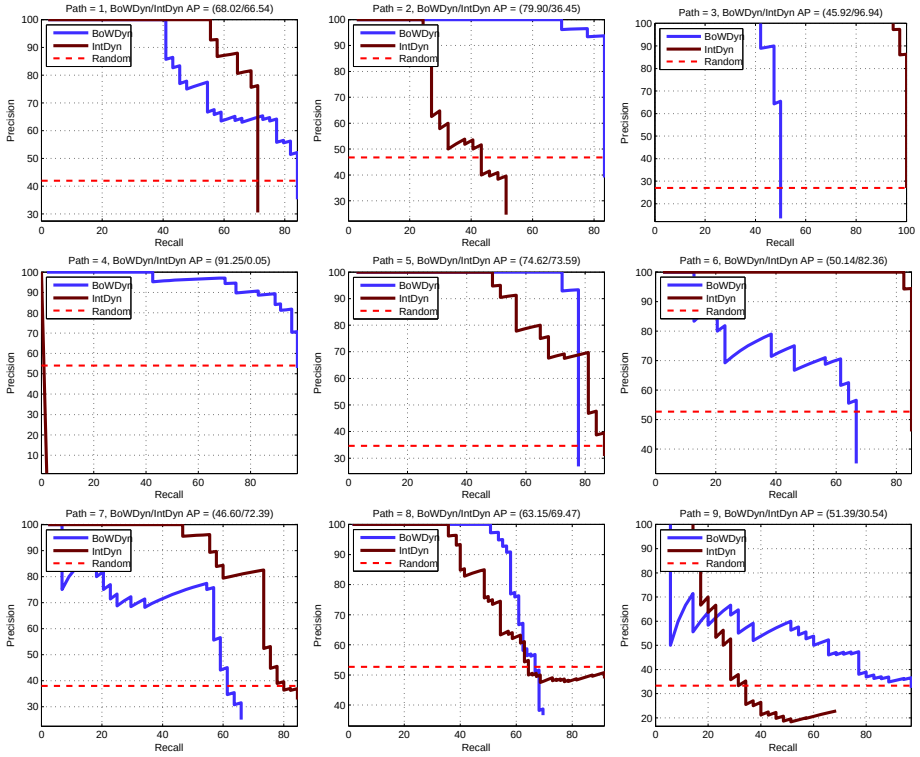


Figure 10: Precision/Recall (PR) curves for the best configuration of BoWDyn (8 states, 400 codewords) and IntDyn (5 states), on all nine search paths.

(randomly sampled from the search paths on which we do not test on in a given cross-validation run). This sample contained half as many *Null* class representatives as there were positive instances. Codebooks of $50, 100$ and 400 words were learned from all descriptors, using k-means++, and used to compute normalized BoW histograms. A C-SVM classifier (learned with LIBSVM [Chang and Lin, 2011]) with an RBF kernel and probability outputs was then trained on the BoW histograms. SVM parameters were optimized via five-fold cross-validation on the training data. For localization, all descriptors within a sliding window were quantized, and the BoW histogram fed to the SVM, resulting in a class prediction and a vector $\boldsymbol{\pi} \in [0,1]^{C+1}$ of posterior class-probabilities. The probability vectors were used for computing PR values. Codebook sizes of $50, 100$ and 400 codewords produced mAPs of 31.3%, 24.9% and 22.4%. This is well below any of the mAP values obtained with either BoWDyn or IntDyn, regardless of the configuration. Only RBF kernels were considered. While other kernels (e.g., a histogram intersection kernel) could potentially improve the SVM results, it is unlikely that they could bridge the large performance gap with respect to the proposed solution. A better explanation for the low mAP values is the limited spatio-temporal information captured by BoW histograms. BoWDyn extends

this mid-level representation with a dynamic modeling component that appears essential to achieve good performance.

6. Discussion

In this work, we have studied the problem of automated localization of target structures in US video. This could be of interest in situations where US examinations need to be performed by non-experts, e.g., in underdeveloped areas of the world or during emergencies when fast access to advanced health care facilities is limited. The main contribution was to show that modeling the dynamics of US video can have significant gains for localization. For this, we have proposed an extension of a recent joint model of video appearance and dynamics, the kernel dynamic texture, so as to capture the dynamics of a BoW representation of probe movement. This enables a reduced memory footprint and localization results competitive, or even superior, to those obtained with intensity information only. The comparison to an equivalent representation without any dynamic modeling showed substantially superior localization performance. We have also introduced a new, annotated, database of US videos, acquired on three different phantoms, designed to evaluate recognition approaches. This facilitates the quantitative evaluation of localization performance, using conventional precision/recall metrics. To the best of our knowledge, this is the first publicly-available US video database of this kind.

While we acknowledge that experiments on phantom data are no substitute for testing on actual patient data, clinical relevance is still high for several reasons. In particular, we argue that the multitude and similarity of noodle structures in our phantoms actually compounds the localization task, due to the inherent ambiguity in noodle constellations. Some of those constellations (e.g., knots) are very similar to each other, particularly in 2D. The fact that our approach still exhibits reasonable localization performance in such cases, gives reason to believe that similar performance could be achieved in a clinical context in which vessels and surrounding structures are typically more unique. To underpin that argument, Fig. 4 shows a synthesis result for a clinical US video, acquired by sweeping an US probe on the liver surface during a radiofrequency ablation of a tumor. While this does not allow any conclusions about localization performance, it at least demonstrates that a KDT-based approach allows to capture the dynamics of clinical US data material.

We emphasize that access to clinical US *images* is not the crucial factor for resorting to a purely phantom-based study, since this data is readily available or could easily be obtained from clinical partners. It is the access to suitable *video* material that inhibits experiments at that point. In fact, it is worth commenting on the multiple aspects of this issue: First, US videos are rarely stored in clinical practice; only images relevant for clinical decisions or treatment are typically archived. Second, US examinations in hospitals are usually performed by experienced physicians, for which localization of structures is straightforward and often involves rapid probe movements or even loss of surface contact. Consequently, even if we assume existence of video data, chances are low that this

data is suitable for processing by our algorithm(s). Asking physicians to follow a specific protocol to ensure good quality videos can only be argued in case the benefits of acquiring this data outweigh the negative aspects for the patient (and physician), such as prolonged examination times. We believe that this study provides enough evidence to support such a request for controlled data acquisition to evaluate our approach on clinical data and, in case of success, in the field.

Another issue worthy of further investigation is the robustness of the proposed localization method. Of particular importance is the robustness to variability of anatomic structures, due to differences among patients and variability of probe orientation/angle. We note that the proposed method leverages a number of properties that are known to enhance this robustness, such as the possibility of acquiring multiple templates per structure and the use of the bag-of-words representation. In general, the number of templates required per structure will be dictated by the degrees of freedom of the imaging process. In many scenarios of interest, however, these are constrained, primarily since the probe is limited to traversing the surface of the human body. Hence, it may be possible to assure good performance with a limited number of templates. This is, for example, the case of the FAST procedure, where we envision a simple examination protocol which restricts probe movement to a roughly regular grid that can be covered by translation and slight tilting. This limits appearance variation and bounds the number of required templates. In any case, the number of templates controls the trade-off between the complexity and accuracy of the proposed localization method. The optimal trade-off is likely to be application specific.

With respect to future improvements, the main limitation of all solutions studied in this work is a delay in localization time, with respect to annotations of human annotators. This is a consequence of the use of a sliding window. All tested approaches require a substantial amount of the sliding window to cover the target before localization can occur. In contrast, human annotators tend to localize the target structures as soon as they intersect the imaging plane.

Other possible improvements are in the area of video representation. While we have adopted a standard spatio-temporal descriptor for the BoW representation of small groups of frames, non-trivial gains could potentially be obtained by developing mid-level features specific to the US image acquisition process. This could, for instance, include features extracted directly from the RF signal, before it is encoded into pixel intensities.

Acknowledgments

This work was partially supported by NSF grants CCF-0830535 and IIS-1208522, as well as NIH/NCI grants 1R01CA138419-0, 2R44CA143234-02A1 and 1R43EB016621.

References

Aylward, S., Jomier, J., Guyon, J.P., Weeks, S., 2002. Intra-operative 3D ultrasound augmentation, in: ISBI.

Bejar, B., Zapella, L., Vidal, R., 2012. Surgical gesture classification from video data, in: MICCAI.

Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R., 2005. Actions as space-time shapes, in: ICCV.

Block, B., 2004. The Practice of Ultrasound: A Step-by-Step Guide to Abdominal Scanning. Thieme. first edition.

Chan, A., Vasconcelos, N., 2005. Probabilistic kernels for the classification of auto-regressive visual processes, in: CVPR.

Chan, A., Vasconcelos, N., 2007a. Classifiying video with kernel dynamic texture, in: CVPR.

Chan, A., Vasconcelos, N., 2007b. Supplementary Material for Classifiying Video with Kernel Dynamic Texture. Technical Report 2007/03. Statistical Visual Computing Laboratory, University of California, San Diego.

Chan, A., Vasconcelos, N., 2008. Modeling, clustering, and segmenting video with mixtures of dynamic textures. Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 30, 909–926.

Chang, C.C., Lin, C.J., 2011. LIBSVM: A library for support vector machines. ACM TIST 2, 1–27.

Chaudhry, R., A.Ravichandran, Hager, G., Vidal, R., 2009. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions, in: CVPR.

Chen, C.Y., Grauman, K., 2012. Efficient activity detection with max-subgraph search, in: CVPR.

Coviello, E., Mumtaz, A., Chan, A., Lanckriet, G., 2012. Growing a bag of systems tree for fast and accurate classification, in: CVPR.

De Cock, K., Moore, B.D., 2000. Subspace angles between linear stochastic models, in: CDC, pp. 1561–1566.

Dollar, P., Rabaud, V., and S. Belongie, G.C., 2005. Behavior recognition via sparse spatio-temporal features, in: VS-PETS.

Doretto, G., Chiuso, A., Wu, Y., Soatto, S., 2003. Dynamic textures. IJCV 51, 91–109.

Duchenne, O., Laptev, I., Sivic, J., Bach, F., Ponce, J., 2009. Automatic annotation of human actions in video, in: ICCV.

Hoai, M., Lan, Z., De la Torre, F., 2011. Joint segmentation and classification of human actions in video, in: CVPR.

Horn, B., Schnuck, B., 1981. Determining optical flow. Artificial Intelligence 17, 185–203.

Kläser, A., Marszalek, M., Schmid, C., 2008. A spatio-temporal descriptor based on 3D-gradients, in: BMVC.

Kläser, A., Marszalek, M., Schmid, C., Zisserman, A., 2010. Human focused action localization in video, in: International Workshop on Sign, Gesture & Activity.

Kwitt, R., Vasconcelos, N., Razzaque, S., Aylward, S., 2012. Recognition in ultrasound: Where Am I?, in: MICCAI.

Laptev, I., Lindeberg, T., 2003. Space-time interest points, in: ICCV.

Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B., 2008. Learning realistic human actions from movies, in: CVPR.

Lv, F., Nevatia, R., 2007. Single view human action recogniton using key pose matching and viterbi path searching, in: CVPR.

Martin, R.J., 2000. A metric for ARMA processes. IEEE Trans. Signal Process. 48, 1164–1170.

Ravichandran, A., Chaudhry, R., Vidal, R., 2009. View-invariant dynamic texture recognition using bag of dynamical systems, in: CVPR.

Sadanand, S., Corso, J., 2012. Action bank: A high-level representation of activity in video, in: CVPR.

Satkin, S., Hebert, M., 2010. Modeling the temporal extent of actions, in: ECCV.

Scalea, T., Rodriguez, A., Chiu, W., Brenneman, F., Fallon, W., Kato, K., McKenney, M., Nerlich, M., Ochsner, M., Yoshii, H., 1999. Focused assessment with sonography for trauma (FAST): Results from an international consensus conference. Journal of Trauma and Accute Care Surgery 46, 466–472.

Sun, D., Roth, S., Black, M., 2010. Secrets of optical flow estimation and their principles, in: CVPR.

Vishwanathan, S., Smola, A., Vidal, R., 2007. Binet-Cauchy kernels on dynamical systems and its application to the analysis of dynamic scenes. IJCV 73, 95–119.

Vondrick, C., Patterson, D., Ramanan, D., 2012. Efficiently scaling up crowdsourced video annotation: A set of best practices for high quality, economical video labeling. IJCV (online).

Ward, J., Lukowicz, P., Gellersen, H., 2011. Performance metrics for activity recognition. ACM TIST 2, 1–23.

Willems, G., Becker, J., Tuytelaars, T., Gool, L.V., 2009. Exemplar-based action recognition in video, in: BMVC.

Willems, G., Tuytelaars, T., Gool, L.V., 2008. An efficient dense and scale-invariant spatio-temporal interest point detector, in: ECCV.