Inferring Unseen Views of People

Chao-Yeh Chen and Kristen Grauman
University of Texas at Austin
chaoyeh@cs.utexas.edu, grauman@cs.utexas.edu

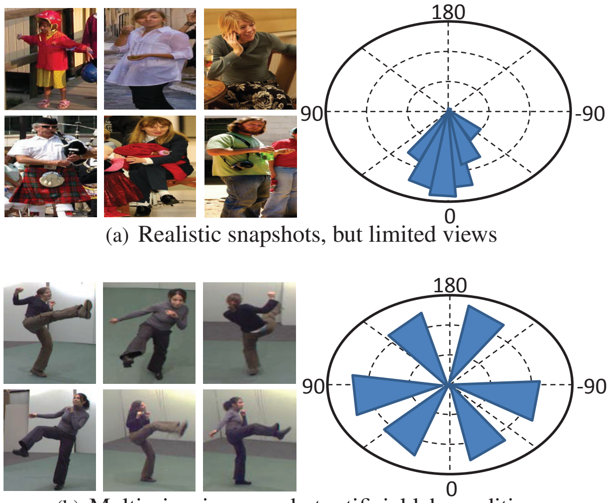
Abstract

We pose unseen view synthesis as a probabilistic tensor completion problem. Given images of people organized by their rough viewpoint, we form a 3D appearance tensor indexed by images (pose examples), viewpoints, and image positions. After discovering the low-dimensional latent factors that approximate that tensor, we can impute its missing entries. In this way, we generate novel synthetic views of people—even when they are observed from just one camera viewpoint. We show that the inferred views are both visually and quantitatively accurate. Furthermore, we demonstrate their value for recognizing actions in unseen views and estimating viewpoint in novel images. While existing methods are often forced to choose between data that is either realistic or multi-view, our virtual views offer both, thereby allowing greater robustness to viewpoint in novel images.

1. Introduction

Analyzing people in images and video is a central problem in computer vision, and it is essential to many applications in surveillance, human-computer interaction, and video indexing. Over the last decade, learning-based methods have made good headway on these challenging problems. A promising paradigm is to extract descriptors of human appearance or motion, and then use supervised learning to predict the parameter of interest—such as the person's activity, orientation, clothing, or identity [33, 17, 11, 39, 42, 5, 44, 34, 22, 6].

In adopting a statistical approach, however, viewpoint sensitivity can be a major stumbling block.[1] A model trained to recognize an activity performed by a forward-facing person will fail when presented with the exact same activity performed by a person viewed from the side—the overall appearance simply will not match. Conscious of this problem, a common approach is to train viewpoint-specific models: using data labeled by both the camera viewpoint and activity class (or other parameter of interest), the system learns what the activity looks like in each of the discrete views [33, 11, 42, 5, 44, 34, 22]; usually this is done only implicitly, by assuming rough viewpoint consistency (e.g., always frontal). Alternatively, given data from multiple cameras simultaneously, some methods learn the statistical connections between viewpoint-specific features and then transfer information between views at test time [8, 13, 18, 20, 45].



(a) Realistic snapshots, but limited views

(b) Multi-view imagery, but artificial lab conditions

Figure 1. The data dilemma for human images. (a) Single view images are often realistic and "unstaged", but populate only a sparse set of camera viewing angles. (b) Multi-view data give full view coverage, but are more artificial in terms of acted poses and simplistic backgrounds. Our method makes use of any available images to envision seen poses in unseen viewpoints.

For any such learning strategy, having training data from a variety of viewpoints is essential. Unfortunately, this is easier said than done. Researchers currently face a data dilemma. On the one hand, Internet images and Hollywood movies offer abundant realistic examples of humans performing various actions, but they are naturally biased towards certain viewpoints (see Figure 1(a)). This is to be expected, since humans tend to take photos of other humans as

[1] Consistent with prior work, and without loss of generality, we focus on *viewpoint* in terms of the camera's azimuth with respect to the human. By *pose* we mean the person's 3D joint configuration due to their action.

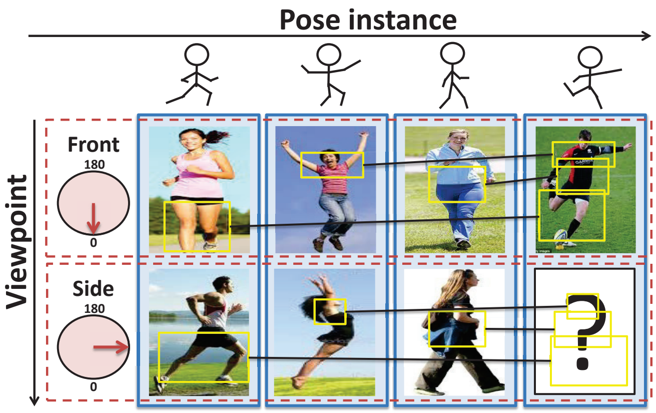


Figure 2. Our approach discovers the latent factors that relate viewpoint and body pose, and uses them to infer unseen views. For example, despite never seeing a kicking pose from any view but frontal (top right image), it hallucinates what it will look like from the side (bottom right). The key is to learn connections between similar looking parts in different poses (here marked with lines for illustration only).

they face the camera. As a result, nice "in the wild" examples are sparse for many other viewpoints, and today's challenge datasets (e.g., PASCAL Actions [7]) are restricted to canonical viewpoints. On the other hand, efforts to collect data specifically from multiple views are prone to scripted behavior and artificial lab environments (see Figure 1(b)). This is also to be expected, since the actors must be instructed to do certain actions while in the special synchronized multi-camera rig.

How can we overcome this dilemma? How can we obtain realistic human image data from varied viewpoints? Rather than physically place more cameras around subjects, our goal is to use whatever viewpoints we *do* have to generate virtual views in those we do not. To this end, we propose a view synthesis approach based on tensor completion. The key idea is to recover the latent factors that relate viewpoint and body pose *without* observing the two neatly varying together—that is, *without observing each pose in all views during training*. This is critical to utilize existing single-view data, but why should it be possible? We observe that from the same viewpoint, people look similar in certain portions of the image, even when they are performing different actions or poses (see Figure 2). Using a latent factor model, we aim to discover these relationships and use them to infer appearance in unseen views.

Our method takes as input images of people organized by their approximate viewpoint. We construct a 3D tensor indexed by the image examples, their viewpoints, and the spatial image positions. Each entry in the tensor records the appearance observed at those coordinates. Notably, many entries are unobserved in the input data. We show that a probabilistic tensor factorization technique can discover the latent factors governing how all three observed dimensions jointly determine appearance. Intuitively, those factors might correspond to things like the type of clothing, body weight, lighting, or partial pose fragments. Using them, we impute missing entries in the tensor, thereby inferring the image descriptors for unobserved views of people that, during learning, may have been observed from just one camera viewpoint.

We show that the inferred views are accurate, which lets us expand existing datasets to fuller viewpoint coverage. Furthermore, we demonstrate the impact for two practical applications. First, we show that our virtual views let the system learn an action category in a viewpoint for which it has never seen any real exemplars, yielding results that are competitive with recent cross-view recognition methods. Second, we show that by using the virtual views to augment real training images, we can predict a person's orientation more accurately in novel images. In both cases, the inferred views help make statistical appearance-based methods robust to viewpoint.

2. Related Work

Image-based rendering for virtual views Existing view synthesis methods originate from image-based rendering [15], where, rather than explicitly construct a 3D model, new views are synthesized directly from multiple 2D views. Typically point correspondences are estimated between views, and then intermediate views are synthesized by warping the pixels appropriately, leveraging insights from projective or multi-view geometry (e.g., [31, 1]). The resulting virtual views can be used to augment training data for object recognition [3], or to reposition the viewpoint at test time [32, 30]. Image-based models of pedestrians using calibrated, synchronized cameras are explored in [32, 12]. Compared to all such methods, our approach to view synthesis relies on *learning*, not geometry and warping. Our method only implicitly captures geometry through its knowledge about discrete viewpoints. This lets us forgo point correspondences, which are difficult to estimate reliably. Furthermore, rather than make strong assumptions about calibrated cameras and/or simultaneous multi-view capture, our method leverages any available views; some instances may appear from as few as one viewpoint.

Synthetic data As an application of our view synthesis idea, we use virtual views to train action and viewpoint detectors. Whereas our virtual views are data-driven, some research exploits graphics engines to create synthetic data for pose estimation [33, 34], action recognition [24], and person detection [26].

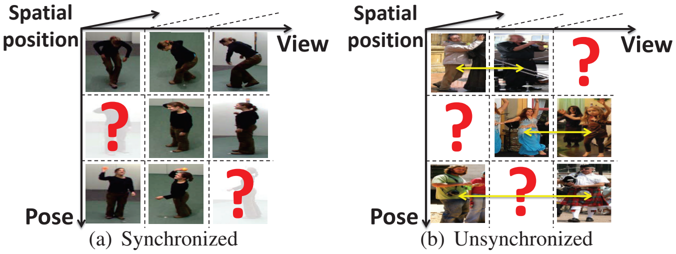
Viewpoint-specific human models Viewpoint-specific models (or mixtures thereof [10]) are common in object recognition (e.g., [35]). Recent methods to recognize actions in static images (e.g., [42, 22, 11, 5, 44]) are also implicitly viewpoint-specific, with some robustness to modest viewpoint changes owing to their use of spatial pooling. Other work develops video features that are robust (even if not strictly invariant) to viewpoint changes [14, 39, 17]. Our contribution is unseen view inference; using our method to expand training sets has the potential to benefit such prior models as well, as we will see in results.

Viewpoint-invariant human models View-invariant methods develop features that remain stable across camera views (e.g., [25, 27, 43, 21]), but they require reliable body joint detection. When multi-view data is available, 3D reconstruction can be used to form 3D exemplars [38] or view-invariant features [41], though their view assumptions and computational demands may be too high for many applications. Multiple action recognition methods *transfer* features between viewpoints, learning the "domain shift" between pairs of views [8, 13, 18, 20, 45]. Such methods require synchronized multi-view data during training, whereas our method can learn from a set of single-view snapshots. Furthermore, we stress that none of these prior methods hallucinate unseen views, as we propose; our method has applications beyond action recognition, including visualization (e.g., helping an artist sketch an actor from a new viewpoint).

Matrix completion Matrix factorization methods are studied extensively for collaborative filtering [23, 16, 28, 29, 40]. Whereas the standard recommender problem can be treated in 2D (items vs. users), our problem has an inherent 3D structure; we account for it using a tensor factorization approach originally developed to model movie ratings as trends vary over time [40]. There is limited work exploring tensor completion for visual data. Existing methods infer missing pixels in a *single* source image/video, e.g., for in-painting [19], or infer new 3D face meshes captured with a structured light scanner for video puppetry [36]. In contrast, our tensor is indexed by intermediate parameters (pose, view) observed across multiple source images, and we explore how inferring unseen images helps recognition.

3. Approach

We pose unseen view inference as a tensor completion problem. Throughout, we consider a set of discrete viewpoints consisting of M orientations of the person with respect to the camera (facing front, front-left, etc.). As input, our method takes cropped images of people organized by their discrete viewpoint ($M = 5$ or 8 in our datasets). As output, our method returns image descriptors capturing the appearance of those same people in each viewpoint from which they were not observed.



(a) Synchronized (b) Unsynchronized

Figure 3. Visualizing the 3D tensor $\mathbf{X}$ in the synchronized (left) and unsynchronized (right) cases. (We display a whole image for visualization purposes, though really its descriptor extends out in the third dimension of the tensor.)

We consider two scenarios: *synchronized* and *unsynchronized*. For the synchronized case, the input images include (at least some) examples of people observed simultaneously by multiple cameras. Any subset of the M views might be present for a given instance, and *the poses in the examples are not annotated in any way* (i.e., no stick figures are given). See Figure 3(a). For the unsynchronized case, the input images are single-view snapshots, such as those one might typically find in online photo collections. See Figure 3(b). In this case, we assume each training image is annotated with body pose (joint positions). In either case, we assume the inputs contain a variety of body poses, though there may be an imbalanced representation of certain poses and viewpoints.

In the following, we first define the tensor and factorization approach for the synchronized case. Then, we generalize it to handle unsynchronized single-view inputs.

3.1. Discovering the Latent Factors

Our model represents human appearance as a function of pose, viewpoint, and position in the image. The goal is to fit a low-dimensional factor model to the observed data, such that the spatially varying appearance can be approximated as a combination of some latent pose and viewpoint factors. As discussed above, the fact that some local appearance patterns re-occur between different poses suggests that such latent factors exist. Intuitively, they might correspond to things like local body configurations (arm outstretched, knee bent, etc.), lighting conditions, or body types.

For each input image, we first extract its K-dimensional appearance descriptor. We use Histograms of Oriented Gradients (HOG) [4], which offer robustness to small shifts and rotations. HOG pools the gradients within a grid of cells, and histograms the pixels per cell into orientation bins; each block of HOG descriptor dimensions originates from a particular spatial region in the image, and adjacent blocks originate from adjacent regions (except for boundary cells). Then, we assign each image to one of the M viewpoints. We currently use ground truth orientation data for this step, as it is available with multiple public datasets [38, 2]; however, automatic methods are also possible, e.g., [22].

Let $i = 1, \ldots, N$ index the input data, where each i corresponds to a unique moment in time—that is, a single snapshot, or a set of multi-view images taken simultaneously. For each of the N inputs, we thus have a descriptor for some number between 1 and M of the total possible viewpoints. Each i captures a distinct pose, whatever pose the human is doing. Thus, we stress that while we refer to the N inputs as "poses", if at least some inputs are multi-view, we do not require pose *annotations* for the input data.

Using this data, we construct a 3D tensor $\mathbf{X} \in \mathbb{R}^{N \times M \times K}$, where entry x_{ij}^{k} corresponds to the image descriptor value in the i-th pose, the j-th view, and the k-th feature dimension (which reflects image position). Let $\mathbf{P} \in \mathbb{R}^{D \times N}$, $\mathbf{V} \in \mathbb{R}^{D \times M}$, and $\mathbf{S} \in \mathbb{R}^{D \times K}$ denote matrices whose columns are the D-dimensional latent feature vectors for each pose, view, and spatial position, respectively. We suppose that x_{ij}^{k} can be expressed as an inner product of latent factors, $x_{ij}^{k} \approx \langle P_i, V_j, S_k \rangle$, where a subscript denotes a column of the matrix. In matrix form, this means $\mathbf{X} \approx \sum_{d=1}^{D} P_{d,:} \circ V_{d,:} \circ S_{d,:}$, where a subscript $d,:$ denotes the d-th row in the matrix, and $\circ$ is the outer product.

To recover the latent factors, we use the Bayesian probabilistic tensor factorization approach of [40], which extends probabilistic matrix factorization [28, 29] to accommodate time-evolving consumer data for movie recommendation tasks. To account for uncertainty, we represent the likelihood distribution for the observed descriptors by

$$p(\mathbf{X}|\mathbf{P}, \mathbf{V}, \mathbf{S}, \alpha) = \Pi_{i=1}^{N} \Pi_{j=1}^{M} \Pi_{k=1}^{K} \left[\mathcal{N}(x_{ij}^{k} | \langle P_i, V_j, S_k \rangle, \alpha^{-1})\right]^{I_{ij}},$$

where $\mathcal{N}(x|\mu, \alpha)$ denotes a Gaussian with mean μ and precision α, and I_{ij} is an indicator variable equal to 1 if pose i appears in view j, and 0 otherwise. We use Gaussian priors for each of the latent factors P_i, V_j, S_k. For pose and viewpoint we use independent Gaussians, while for the spatial factors we use the prior $S_k \sim \mathcal{N}(S_{k-1}, \Sigma_S)$, for $k = 2, \ldots, K$, which reflects that descriptor values are likely to vary smoothly in spatially close regions.[2] Let Θ denote a set of random variables comprised of the mean and covariance of all three factors, including Σ_S. For all Gaussian prior hyper-parameters (α and the variables in Θ), we use conjugate distributions as priors to facilitate subsequent sampling steps.

Following [29, 40], we integrate out all the model parameters and hyper-parameters to obtain a predictive distribution for an unseen view given all observed input images:

$$p(\hat{x}_{ij}^{k}|\mathbf{X}) = \int p(\hat{x}_{ij}^{k}|P_i, V_j, S_k, \alpha) p(\mathbf{P}, \mathbf{V}, \mathbf{S}, \alpha, \Theta|\mathbf{X}) \; d\{\mathbf{P}, \mathbf{V}, \mathbf{S}, \alpha, \Theta\}.$$

Compared to solving for a single point estimate for the MAP factors $\mathbf{P}^*$, $\mathbf{V}^*$, $\mathbf{S}^*$, this helps prevent overfitting to poorly tuned hyper-parameters. It is approximated using Markov chain Monte Carlo (MCMC) sampling:

$$p(\hat{x}_{ij}^{k}|\mathbf{X}) \approx \sum_{l=1}^{L} p(\hat{x}_{ij}^{k}|P_i^{(l)}, V_j^{(l)}, S_k^{(l)}, \alpha^{(l)}), \qquad (1)$$

where L denotes the number of samples. The samples $\{P_i^{(l)}, V_j^{(l)}, S_k^{(l)}, \alpha^{(l)}\}$ are generated with Gibbs sampling on a Markov chain whose stationary distribution is the posterior over the model parameters and hyper-parameters $\{\mathbf{P}, \mathbf{V}, \mathbf{S}, \alpha, \Theta\}$. Sampling is initialized using the MAP estimates of the three factor matrices. See [40] for details.

With this tensor formulation, we capture the global influence that image position has on all the poses and viewpoints, which is very informative for cropped person images. For example, the model can learn that the presence of strong -45 degree gradients in cells in the bottom right of the person bounding box when viewed from the front (due to an extended left leg) suggests the likely presence of 45 degree gradients within the associated bottom left cells if he were viewed from behind.

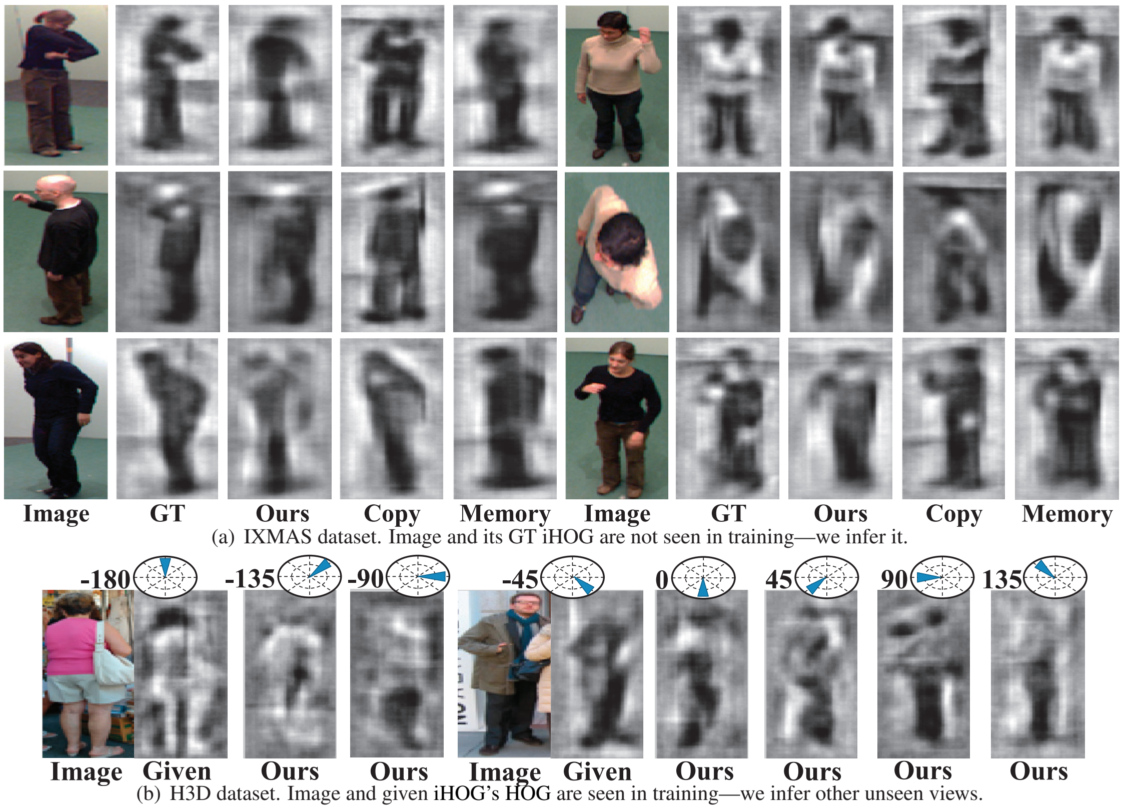
We choose to infer descriptors, rather than raw pixels. The gradient-based HOGs offer robustness to low-level appearance differences (e.g., clothing), such that we can expect to learn latent factors with less input data than would be needed for raw pixels. Inferring pixel intensities, though in principle possible with the same approach, would likely waste modeling effort on unneeded detail (a typical person bounding box in our datasets contains 6,000 pixels, but only 108 HOG dimensions). In addition, as we demonstrate below, we can use the inferred views directly in later learning tasks, since most vision methods operate in a feature space other than pixels. Plus, to visualize the results, we can "invert" HOG descriptors back into image space with [37].

3.2. Learning with Unsynchronized Single-View Images

Next we generalize our approach to handle the challenging case where only unsynchronized single-view data is available. Doing so will allow us to exploit existing realistic data sources, such as photos on Flickr. Presumably humans can infer unseen views because they have seen many individuals in various poses and viewpoints, not because they have seen carefully orchestrated multi-view examples for individual people. They understand the pose associations across individuals. In a similar vein, our idea is to link snapshots that contain *similar* 3D body poses, but *different* viewpoints. In this way, a pose "instance" in the tensor can be comprised of different individual people (as depicted in Figure 3(b)).

This variant requires pose-labeled training data, using either manual or automatic annotations. Good tools are

[2] Accounting separately for the boundary cells (which need not be smooth a priori) would add complexity to the model, and we find it is sufficient in practice not to.



(a) IXMAS dataset. Image and its GT iHOG are not seen in training—we infer it.

(b) H3D dataset. Image and given iHOG's HOG are seen in training—we infer other unseen views.

Figure 4. Visualization of inferred views using inverted HOGs. Best viewed on pdf.

available to semi-automate pose labeling [2], making this requirement manageable.

Let $\boldsymbol{p}_q \in \mathbb{R}^{3J}$ denote the normalized body pose configuration for image q. Its $3J$ elements are the 3D positions of J body joints, normalized to a common coordinate system where they can be meaningfully compared. Specifically, we shift the raw skeleton to place the center of the hips at the origin, rotate it to align the plane connecting the hips and neck to be orthogonal to the z axis, and scale it to the average head-to-toe height. We estimate the pose distance between two images as $d(q, r) = ||\boldsymbol{p}_q - \boldsymbol{p}_r||_2$. Then we sort all training pairs by $d(q, r)$, and take any pairs whose pose distance is less than 0.2 times the average distance. Each such pair provides two K-dimensional HOG entries for the tensor, placed at the appropriate two columns based on their viewpoints.[3] Once the linked pairs are entered into the tensor, we perform inference as described above.

With this extension, even if an "in the wild" snapshot was observed from just a single viewpoint, we can infer its appearance in novel views. As such, our method provides downstream estimation tasks (e.g., action recognition) with data that is both more complete *and* realistic. Furthermore, while our current implementation focuses on the multi-view and single-view cases separately, our approach naturally supports a mix of both types of data. In that case, the algorithm will learn the multi-view constraints from synchronized instances and propagate them to single-view instances during inference.

4. Experimental Results

We validate our approach on two public datasets. The first, INRIA Xmas Motion Acquisition Sequences (IXMAS) [38], contains multi-view synchronized data from $M = 5$ cameras, with 11 actions (check watch, cross arms, kick, etc.) performed by 10 actors, for 16,800 total images. The second, Humans in 3D (H3D) [2], contains 2,378 single-view Flickr images, with people doing various unscripted poses (reaching, walking, riding a bike, etc.), and has 3D pose annotations for $J = 33$ joints done by MTurkers. We use the viewpoint annotations of [22].

We extract HOG with 9 cells and 12 bin histograms per cell, yielding a $K = 108$ dimensional descriptor per image. We use the factorization code of [40], and fix the latent factor dimensionality to $D = 500$ and the number of samples $L = 500$, based on cross-validation on training data, and $\alpha = 2$ as default. We clip inferred outputs to $[0, 1]$, the valid HOG range. With these parameters, and with $N = 2,200$ instances, learning the latent factors takes about 6 hours. Inferring feature values requires only two inner products, which takes < 1 ms.

[3] Preliminary tests in which we link beyond pairs of examples did not show a noticeable difference in results.

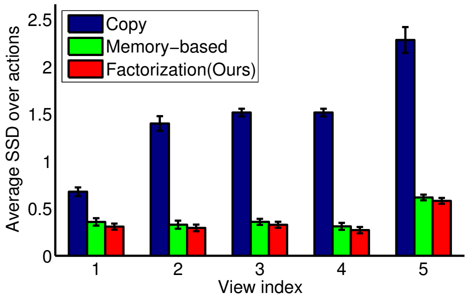


Figure 5. Error in inferred views

We evaluate how well our inferred views match the (withheld) ground truth images. In addition, we compare to a variety of state-of-the-art view-invariant recognition methods as well as two baseline techniques for virtual view creation: 1) Memory, a memory-based tensor completion approach and 2) Copy, a method that copies observed images from nearby views. For Memory, we adapt a neighborhood approach in collaborative filtering [16] to our problem setting (see Supp.). For Copy, we find the observed image in the training data *for the very same pose instance* that is nearest in viewpoint to the desired unseen view, and copy its HOG descriptor. For example, if the needed view j were frontal, and the view 45 degrees off of frontal appears in the training set, that would be the estimate. Note that a traditional warping approach is inapplicable for these tests, since it demands multi-view calibrated data, and can warp only to fairly nearby views (i.e., not ground to overhead).

In the following, we first evaluate the inferred views' accuracy (Sec. 4.1). Then we use the virtual views for two applications: action recognition (Sec. 4.2) and viewpoint estimation (Sec. 4.3).

4.1. Accuracy of Inferred Views

Figure 4 visualizes inferred views using the "HOG goggles" inverted-HOG (iHOG) technique, which inverts a HOG descriptor back to a natural image [37]. Here we use HOG descriptors with higher dimension (90 cells×12 bins =2970) to provide detailed visualization. We compare the view inferred by our method to the iHOG for the real ground truth (GT) image, which is the upper bound on quality. The two often look quite similar, which means our method infers the true appearance well. While Copy's results can look realistic—after all, they originate from HOGs on real images—they are not as accurate as ours. This underscores the value in modeling the latent factors for all observations, rather than simply matching to the nearest available view. Our advantage is most striking in the most difficult cases, such as inferring the overhead view (middle row, right side of (a)). For poses that appear similar between views (bottom row, left side of (a)), Copy is competitive, as expected. The H3D visualizations (b) are noisier due to fewer observed features and cluttered backgrounds, yet we still capture the shape of the person and some articulated details of the pose (e.g., see the bent arm in far right). (Note, on H3D Copy simply returns the given iHOG for all other views.) See Supp. for more examples.

Copy	Memory	Ours	Ground truth
15.08 (2.45)	20.39 (2.49)	**34.32 (3.47)**	60.36 (2.51)

Table 1. Action recognition accuracy (mAP) in an unseen viewpoint on IXMAS. Numbers in parens are standard errors.

Figure 5 quantifies these observations. We randomly sample 200 images for each action in IXMAS, for a total of 2,200 images. Then for each action in turn, we withhold all images for that action in a given view, apply factorization, and compare the inferred unseen views to the withheld ground truth. We plot the Summed Square Difference (SSD) error between inferred and actual views, for each view in IXMAS. (H3D lacks the ground truth to make this evaluation possible.) Our factorization method outperforms both baselines. As to be expected, view 5, the overhead view, is most difficult for all methods; nonetheless, our inferred views remain 74% better than Copy and 6% better than Memory.

These results validate the main goal of our approach: to accurately map seen poses to unseen views, even when training examples are single-view, asynchronous, and captured in complex environments. In the remaining results, we will further demonstrate that having estimated the unseen views well, we are better positioned to train viewpoint-sensitive models for recognition tasks.

4.2. Recognizing Actions in Unseen Views

Next, we use our inferred views to train a system to recognize actions from a viewpoint it never observed in the training images. As above, for each IXMAS action label in turn, we hold out all its images in a given viewpoint, and then infer the unseen views. We use those inferred HOGs to train a viewpoint-specific one-vs.-rest SVM action classifier for that action category; the positive exemplars are all synthetic, while the negative exemplars are real images from all other action labels. We evaluate accuracy on a test set of single-view static images consisting of 200 real positives and 2000 real negatives.

Table 1 shows the results. Our method significantly outperforms the baselines. Compared to Memory, our recognition advantage is much greater than our SSD advantage in Figure 5, which suggests the perceptual quality differences are greater than what SSD captures. We also show an upper bound—the accuracy that would be obtained if the *real* images had been available, rather than inferred ("Ground truth"). Naturally, the accuracy is higher using real training images; still, we more than double the accuracy of a method that uses the nearest available real view (Copy).

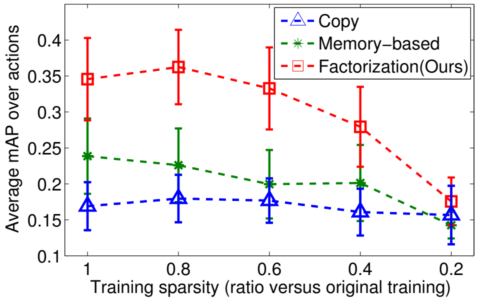


Figure 6. Accuracy in unseen views as a function of tensor sparsity.

No occlusions	Occluded training	Partially visible testing
37.7 (3.06)	36.9 (3.03)	52.6 (2.07)

Table 2. Testing the impact of occlusions (average mAP)

Figure 6 evaluates the impact of input data sparsity. We repeat the recognition task above, but now with an increasingly sparse set of real input views for training. To increase sparsity, we remove views at random. Our method's accuracy is fairly stable up until about 40% (i.e., when 60% of the tensor is unobserved), showing the power of the latent factors with rather incomplete data. While our accuracy starts to decline when the observed features comprise less than half of the tensor entries, it is still substantially better than the baselines. With only 20% observed data, all methods do similarly, indicating insufficient information about the feature correlations between the views. COPY's standard error increases with sparsity; it suffers once fewer nearby views are available.

Next, we demonstrate how our method can infer missing views in the face of partial occlusions. Table 2 shows the results, for action recognition on the first five IXMAS actions. The columns compare our method's accuracy in three scenarios: 1) with no occlusions, 2) when training examples are partially occluded, and 3) when test examples are partially visible. To generate the training set occlusions, we randomly remove 20% of the HOG cells; to generate the test set occlusions, we omit the lower body region. Comparing columns 1 and 2, we see our method maintains its accuracy in spite of occluded training examples, showing the latent factors have a similar effect for missing data within an image, not just within the viewpoints. Comparing columns 1 and 3, we see that if the unobserved views are partially visible, our method can even more precisely complete them.

Finally, we use our inferred views to compare to several existing methods for cross-view action recognition. We follow the standard leave-one-action-out IXMAS protocol [8]. We train an action class using the HOG features from all frames, and predict the action label of a test clip by voting. Table 3 shows the results. They are quite encouraging. Despite using a rather simple frame-based HOG classifier, our inferred views lead to recognition accuracy better than four existing methods that devise sophisticated features or learning algorithms specifically for this recognition task. This shows that explicitly estimating missing views can offer advantages over using view-invariant descriptors. That said, we do underperform two of the methods. We suspect our static frame HOG representation is a handicap, as the other methods use temporal features. It will be interesting future work to generalize our idea to the temporal domain.

	View 0	View 1	View 2	View 3	View 4
Farhadi 08 [8]	61	67	61	63	40
Junejo 08 [14]	63.0	64.3	64.5	58.9	46.6
Farhadi 09 [9]	74	77	76	73	72
Liu 11 [20]	79.0	74.7	75.2	76.4	71.2
Li 12 [18]	83.4	79.9	82.0	85.3	75.5
Zhang 13 [45]	**88.3**	**83.0**	**87.7**	**88.3**	**81.9**
COPY	59.9	56.5	53.4	59.8	41.2
MEMORY	67.7	63.0	58.6	65.0	48.9
Ours	79.9	80.8	79.0	80.2	74.2

Table 3. Cross-view action recognition accuracy on IXMAS

(a) Average mAP, compared to view synthesis baselines

Orig	Orig+COPY	Orig+MEMORY	Orig+Ours
17.29	14.77	19.94	**20.30**

(b) Classification accuracy vs. state-of-art

Poselet activations+SVM [22]	Ours
48.4%	**49.9%**

Table 4. Viewpoint estimation accuracy on H3D when we augment real training images with inferred views, compared to alternative view synthesis methods (a) and a state-of-the-art technique (b).

On top of its good performance on this specific task, our method offers functionality the prior work does not: 1) it can translate seen images to images in new viewpoints, whereas the prior methods produce invariant features, which cannot be used in support of other prediction tasks, and 2) it can leverage any available views during learning, whereas the prior methods focus on learning connections only between pairs of views.

4.3. Estimating Body Orientation

Next we test our unsynchronized method (Sec. 3.2) on H3D. We quantize the torso orientations into $M = 8$ discrete views. We use views inferred by our method to augment a training set of real images, then learn viewpoint classifiers. We form a 75%-25% train-test split, and balance the training images per view, since highly imbalanced training images would favor our approach. We train SVMs with χ^2 kernels for all methods. Given a novel test image, we need to decide which way the person is facing. Table 4(a) shows the mAP results. Adding the view-specific training instances created by our method, accuracy is better than training with the real images alone. Furthermore, our factorization approach is again stronger than both baselines.

Next, we compare our viewpoint estimation to an existing method based on poselets [22]. We use the same features, classifier, and experimental setup described in that paper. We train one classifier with the real H3D images, and another with those same images plus our inferred views. Table 4(b) shows the classification accuracy results.[4] We see our virtual views boost the accuracy of this state-of-the-art approach for viewpoint estimation.

Both these H3D results are encouraging. Not only can we infer how a person will appear in other viewpoints having seen him in only a single view, but doing so improves robustness for appearance-based viewpoint estimation.

5. Conclusions

We presented a novel approach for inferring human appearance in unseen viewpoints. Whereas existing methods tackle the problem using geometry and image warping, we offer a new perspective based on learning. We show how to cast the problem in terms of tensor completion, and adapt a factorization approach to accommodate both synchronized and unsynchronized single-view images. Our results on two challenging datasets show that not only can we infer unseen views, but that doing so is useful for practical human analysis tasks. In future work, we plan to extend our idea to handle video data and infer appearance over time.

Acknowledgements This research is supported in part by DARPA CSSG.

References

[1] S. Avidan and A. Shashua. Novel view synthesis in tensor space. In *CVPR*, 1997.

[2] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3D human pose annotations. In *ICCV*, 2009.

[3] H. Chiu, L. Kaelbling, and T. Lozano-Perez. Virtual training for multi-view object class recognition. In *CVPR*, 2007.

[4] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *CVPR*, 2005.

[5] V. Delaitre, J. Sivic, and I. Laptev. Learning person-object interactions for action recognition in still images. In *NIPS*, 2011.

[6] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *PAMI*, 34, 2012.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The Pascal visual object classes (VOC) challenge. In *IJCV*, volume 88, pages 303–338, 2010.

[8] A. Farhadi and M. Tabrizi. Learning to recognize activities from the wrong view point. In *ECCV*, 2008.

[9] A. Farhadi, M. K. Tabrizi, I. Endres, and D. A. Forsyth. A latent model of discriminative aspect. In *ICCV*, 2009.

[10] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[11] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Pose search: retrieving people using their pose. In *CVPR*, 2009.

[12] K. Grauman, G. Shakhnarovich, and T. Darrell. Inferring 3D structure with a statistical image-based shape model. In *ICCV*, 2003.

[13] C.-H. Huang, Y.-R. Yeh, and Y.-C. Wang. Recognizing actions across cameras by exploring the correlated subspace. In *ECCV*, 2012.

[14] I. Junejo, E. Dexter, I. Laptev, and P. Perez. Cross-view action recognition from temporal self-similarities. In *ECCV*, 2008.

[15] S. B. Kang. A survey of image-based rendering techniques. In *Videometrics SPIE Intl Symp on Elec Imag: Science and Technology*, 1999.

[16] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009.

[17] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[18] R. Li and T. Zickler. Discriminative virtual views for cross-view action recognition. In *CVPR*, 2012.

[19] J. Liu, P. Musialski, P. Wonka, and J. Ye. Tensor completion for estimating missing values in visual data. In *ICCV*, 2009.

[20] J. Liu, M. Shah, B. Kuipers, and S. Savarese. Cross-view action recognition via view knowledge transfer. In *CVPR*, 2011.

[21] Q. Liu and X. Cao. Action recognition using subtensor constraint. In *ECCV*, 2012.

[22] S. Maji, L. Bourdev, and J. Malik. Action recognition from a distributed representation of pose and appearance. In *ICCV*, 2011.

[23] B. Marlin. Modeling user rating profiles for collaborative filtering. In *NIPS*, 2003.

[24] P. Matikainen, R. Sukthankar, and M. Hebert. Feature seeding for action recognition. In *ICCV*, 2011.

[25] V. Parameswaran and R. Chellappa. View invariance for human action recognition. *IJCV*, 66(1):83–101, 2006.

[26] L. Pishchulin, A. Jain, C. Wojek, T. Thormaehlen, and B. Schiele. In good shape: Robust people detection based on appearance and shape. In *BMVC*, 2011.

[27] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *IJCV*, 50(2):203–226, 2002.

[28] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *NIPS*, 2007.

[29] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *ICML*, 2008.

[30] S. Savarese and L. Fei-Fei. View synthesis for recognizing unseen poses of object classes. In *ECCV*, 2008.

[31] S. Seitz and C. Dyer. View morphing. In *SIGGRAPH*, 1996.

[32] G. Shakhnarovich, L. Lee, and T. Darrell. Integrated face and gait recognition from multiple views. In *CVPR*, 2001.

[33] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *ICCV*, 2003.

[34] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from a single depth image. In *CVPR*, 2011.

[35] P. Viola and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. In *CVPR*, 2001.

[36] D. Vlasic, M. Brand, H. Pfister, and J. Popović. Face transfer with multilinear models. *ACM Tran on Graphics*, 24(3):426–433, 2005.

[37] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba. HOGgles: Visualizing object detection features. In *ICCV*, 2013.

[38] D. Weinland, E. Boyer, and R. Ronfard. Action recognition from arbitrary views using 3D exemplars. In *ICCV*, 2007.

[39] D. Weinland, M. Ozuysal, and P. Fua. Making action recognition robust to occlusions and viewpoint changes. In *ECCV*, 2010.

[40] L. Xiong, X. Chen, T. Huang, J. Schneider, and J. Carbonell. Temporal collaborative filtering with Bayesian probabilistic tensor factorization. In *SDM*, 2010.

[41] P. Yan, S. Khan, and M. Shah. Learning 4D action feature models for arbitrary view action recognition. In *CVPR*, 2008.

[42] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *CVPR*, 2010.

[43] B. Yao and L. Fei-Fei. Action recognition with exemplar based 2.5D graph matching. In *ECCV*, 2012.

[44] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and L. Fei-Fei. Action recognition by learning bases of action attributes and parts. In *ICCV*, 2011.

[45] Z. Zhang, C. Wang, B. Xiao, W. Zhou, S. Liu, and C. Shi. Cross-view action recognition via a continuous virtual path. In *CVPR*, 2013.

[4] Note that the numbers in (a) and (b) are not comparable to each other due to differences in features and experimental setup.