Who's Better? Who's Best? Pairwise Deep Ranking for Skill Determination

Hazel Doughty Dima Damen Walterio Mayol-Cuevas
University of Bristol, Bristol, UK
<Firstname>.<Surname>@bristol.ac.uk

Abstract

This paper presents a method for assessing skill from video, applicable to a variety of tasks, ranging from surgery to drawing and rolling pizza dough. We formulate the problem as pairwise (who's better?) and overall (who's best?) ranking of video collections, using supervised deep ranking. We propose a novel loss function that learns discriminative features when a pair of videos exhibit variance in skill, and learns shared features when a pair of videos exhibit comparable skill levels. Results demonstrate our method is applicable across tasks, with the percentage of correctly ordered pairs of videos ranging from 70% to 83% for four datasets. We demonstrate the robustness of our approach via sensitivity analysis of its parameters.

We see this work as effort toward the automated organization of how-to video collections and overall, generic skill determination in video.

1. Introduction

How-to videos on sites such as YouTube and Vimeo, have enabled millions to learn new skills by observing others more skilled at the task. From drawing to cooking and repairing household items, learning from videos is nowadays a commonplace activity. However, these loosely organized collections normally contain a mixture of contributors with different levels of expertise. The querying person needs to decide who is better and who to learn from. Furthermore, the number of *how-to* videos is only likely to increase, fueled by more cameras recording our daily lives. An intelligent agent that is able to assess the skill of the subject, or rank the videos based on the skill displayed, would enable us to delve into the wealth of this on-line resource.

In this work, we attempt to determine skill for a variety of tasks from their video recordings. We base this work on two assumptions, first - for tasks where human observers *consistently* label one video as displaying *more skill* than another, there is enough information in the visual signal to automate that decision; and second - the same framework for determining skill can be used for a variety of tasks ranging from surgery to drawing and rolling pizza dough.

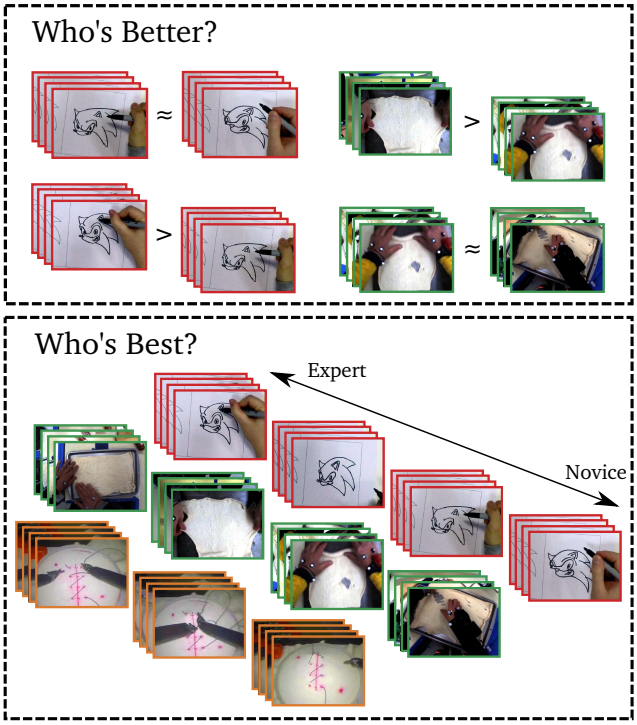

Figure 1. Determining skill in video. **Who's Better?** (Top): pairwise decisions of videos containing the same task, performed with varying or comparable levels of skill. **Who's Best?** (Bottom): ranking learned from pairwise decisions.

We propose to determine skill using a pairwise deep ranking model, which characterizes the difference in skill displayed between a pair of videos, where one is ranked higher than the other *by human annotators* (Fig 1). We use a Siamese architecture where *each stream* is made up of a two-stream (spatial and temporal) convolutional neural network (2S-CNN). This Siamese architecture is trained using a novel ranking loss function that considers the extent of the task within the video, and includes pairs of videos where the skill level is indistinguishable. By assigning videos a relative score of skill for the given task, we can predict a *skill ranking* for a set of videos.

Our main contributions are as follows: i) We present the first method to determine skill in videos for a wide variety

of tasks. ii) We propose a novel ranking loss function which considers the extent of the video and incorporates pairwise similarities in training. This loss function outperforms the standard ranking loss on all datasets by up to 5%. iii) We present pairwise skill annotations for three datasets, two of which are newly recorded. iv) We evaluate our approach on four datasets (two public); one surgical - for which there is authoritative expert ranking, another on rolling pizza dough, as well as two newly introduced datasets for the tasks of drawing and using chopsticks. Newly recorded datasets and annotations are available from the authors' webpages.

2. Related Work

In this section, we review skill determination works in video, primarily for surgical tasks and within sports. We relate this work to the new surge for utilizing collections of *how-to* instructional videos. Finally, we introduce deep ranking approaches, on which our method is based.

Skill Determination. There have been few prior works on automatically determining skill from video. The majority of these works are focused on surgical tasks [20, 27, 28, 36, 37, 39, 40, 41], due to the intensive training needs in this area. For instance, Sharma *et al*. [27] use motion textures to predict the OSATS criteria: a measure of skill specific to the surgical domain. In [39], Zia *et al*. rely on the repetitive nature of surgical tasks, using the entropy of repeated motions to identify different skill levels. Malpani *et al*. [20] use a combination of video and kinematic data to rank performance in two surgical tasks. However, they decompose each task into a sequence of actions, and design specific features for performance evaluation of surgical maneuverer, which makes this inapplicable to non-surgical tasks. Generally, the high specialty of the tasks and methods involved in surgery make these approaches difficult to generalize.

Many of these methods take a coarse approach to identifying skill, splitting participants into categories of novice and expert [40]. Often skill labels are determined by participants' previous experience, instead of their performance in individual videos [11, 40]. We aim however, to rank the performance in each video, instead of classifying the video, or all of a participant's videos, as expert or novice.

A work that utilizes ranking for surgical tasks is that of Zhang *et al*. [37]. It uses relative Hidden Markov Models to evaluate human motion skill by obtaining a ranking between pairs. This work is somewhat limited by the ground truth data: the assumption is that a video recorded at a later date will capture better performances than a participant's earlier recording. Thus, skill is only compared within a participant's performances.

There is also some skill assessment work in the domain of sport [4, 7, 12, 17, 23, 24, 25]. However, many of these works are not generalizable to domains outside sports as they either craft features specific to a sport, such as basketball [4, 17], or focus on quality of motion [7, 12, 23, 24]. The most relevant of these works is from Pirsiavash *et al*. [25], who present a general method for assessing the quality of actions. This is done by estimating human body pose with a skeleton model in order to predict the score of actions, again in sports videos. However, quality of motion on its own is not an essential condition to determine skill. For example, moving a brush in an artistic manner is not a sufficient measure for painting skills.

How-To Videos. Related to skill determination are works on instructional videos [2, 8, 3, 18], that study videos of different people performing the same activity. However, none of these works determine skill from these videos, focusing instead on aligning the steps undertaken to complete the task [2] and the object states and manipulations that occur during the task [3]. Kim et al. [18] evaluate the semantic similarity of action units to determine if two people are performing the same sub-activity, however this is not capable of assessing the skill within the same task or sub-tasks.

Deep Ranking. The most widely used learning to rank formulation is pairwise ranking. The method aims to minimize the average number of incorrectly ordered pairs of elements in a ranking, by training a binary classifier to decide which element in a pair should be ranked higher. This formulation was used by Joachims in RankSVM [15], where a linear SVM is used to learn a ranking. It was originally used to learn search engine retrieval functions from click-through data, however it has been adopted in other ranking applications, including ranking relative attributes in images [22].

Pairwise ranking has also been used in deep learning, first by Burges et al. [5] with RankNet. For instance, Yao *et al*. [33] use a pairwise deep ranking model to perform highlight detection in egocentric videos using pairs of highlight and non-highlight segments. They use a ranking form of hinge loss as opposed to the binary cross entropy loss used in RankNet. In our paper we base our ranking loss on the pairwise margin loss used by Yao *et al*., but with several novel additions, including a pairwise similarity loss.

Other non-pairwise methods for deep ranking such as list-wise ranking [6] have been proposed, yet are less frequently used compared to the pairwise approach, unless optimizing for a specific evaluation metric such as NDCG [14].

3. Learning to Determine Skill

In this section we first give an overview of the skill determination problem and the Siamese two-stream CNN architecture we use to determine skill. We then present our novel additions to the pairwise margin loss function used to train both streams of the CNN.

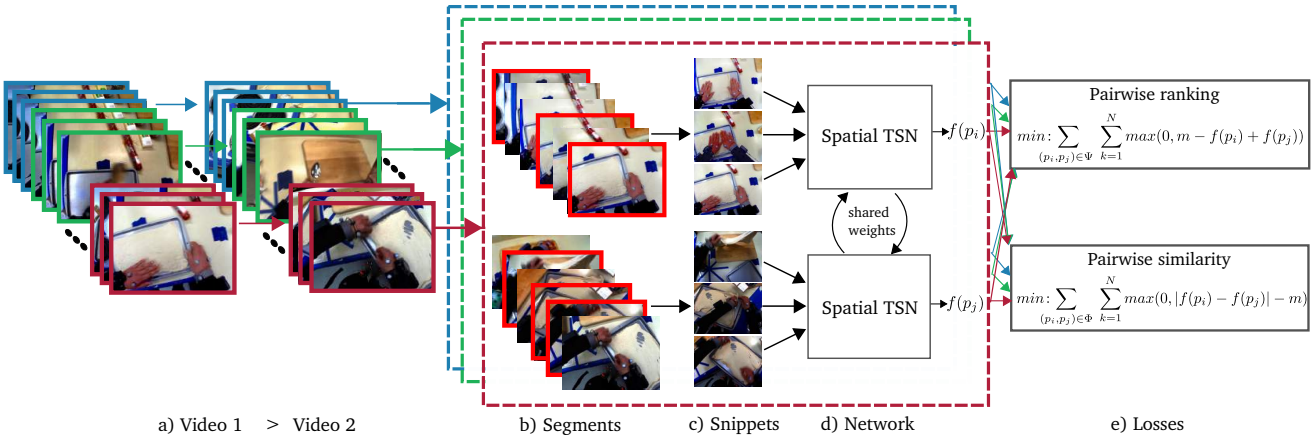

Figure 2. Training for skill determination. a) We consider all pairs of videos, where the first is showing a higher level of skill Ψ, or their skill is comparable Φ, and divide these into N splits to make use of the entire video sequence. b) Paired splits are then divided up into 3 equally sized paired segments as in [32]. c) TSN selects a snippet randomly from each segment. For the spatial network this is a single frame, for the temporal network this is a stack of 5 dense horizontal and vertical flow frames. d) Each snippet is fed into a Siamese architecture of shared weights, for both spatial and temporal streams, of which only the spatial is shown here. e) The score from each split is either fed to the proposed loss functions: ranking/similarity which compute the margin ranking loss based on the pair's label.

3.1. Problem Definition

Our goal is to learn models for ranking skill in different tasks. Given a task, we have a set of K videos $P = \{p_k, 1 \leq k \leq K\}$, from multiple people, each performing the task one or more times. We consider each video independently, as people differ in the skill they display in each video, even across multiple runs. We are thus interested in ranking relative skill per video instead of accumulating a score per person.

$$E(p_i, p_j) = \begin{cases} 1 & p_i \text{ shows higher skill than } p_j \\ -1 & p_j \text{ shows higher skill than } p_i \\ 0 & \text{no skill preference} \end{cases} \quad (1)$$

Note that according to Eq. 1, $E(p_i, p_j) = -E(p_j, p_i)$, we thus need to only obtain one annotation for each pair. We explain how these annotations are obtained in Section 4.1.

3.2. Time as a Measure of Skill

A naive way to approach measuring skill is to use time of completion, as finishing a task faster (or slower) could imply a higher level of skill. However, from the JIGSAWS dataset [11] we prove that time is not sufficient. Although there is some correlation between score and time in the Knot Tying task ($\rho = 0.72$), there is little correlation in the Needle Passing ($\rho = 0.23$) and Suturing ($\rho = 0.34$) tasks. Therefore, although time can be useful in some tasks, it is not a general or reliable method for skill determination. We thus propose a method for skill determination that is independent of time of completion.

3.3. Temporal Segment Networks as Architecture

Tasks differ in how skill can be demonstrated. In this respect we identify two main sources of relevant information. The first is the quality and type of motions used. The second is the effect on the environment captured through the appearance of the task. We thus utilize two stream convolutional neural networks (2S-CNN) for skill determination. Specifically, we base our method on Temporal Segment Networks (TSN) [32]. We select TSN due to their state of the art performances on action recognition benchmarks and ability to model long range temporal structure and dependencies.

In training TSN, as in [32], we uniformly divide each input video sequence into three segments, then randomly sample a single short snippet from each of these segments (Fig 2b,c). For each iteration in training, our 2S-CNN outputs a preliminary prediction of skill for each snippet. This decision is then pooled across the three snippets, creating a score per input video. The output to the loss function (Fig. 2e), in both the spatial and temporal streams, is then the consensus between selected snippets.

3.4. Pairwise Deep Ranking

We use the pairwise approach for learning to rank. To do this we build a Siamese version of the two-stream TSN described in Section 3.3, with the weights shared across both sides of the Siamese network (Fig. 2d). Given a pair of videos, where the first video is ranked higher than the second in terms of skill, we want the Siamese network to output a higher score for the first. Formally, we have a set of pairs $\Psi = \{(p_i, p_j); E(p_i, p_j) = 1\}$ (ref Eq. 1). These two

videos are fed into the separate, but identical, TSNs which form the Siamese network (Fig 2a). Assuming the TSN outputs $f(\cdot)$, our goal is to learn the function f such that we determine skill, where

$$f(p_i) > f(p_j) \quad \forall (p_i, p_j) \in \Psi \tag{2}$$

To gain an overall rank for all videos, we use a margin loss layer to evaluate the loss for each pair. The loss function we use is an approximation to 0-1 ranking error loss that has been used successfully for other applications [33, 31];

$$L_{rank1} = \sum_{(p_i,p_j)\in\Psi} max(0, m - f(p_i) + f(p_j)) \tag{3}$$

We use $m = 1$ in our experiments. During training, this loss function evaluates the violation of the ranking of each pair of videos and back-propagates the gradient through the network. This allows the network to learn discriminative features to distinguish between the amount of skill displayed in different videos.

3.5. Pairwise Deep Ranking with Splits

Traditionally, 2S-CNN are used for action recognition [29], thus the whole length of the video needs to be considered once to recognize the undertaken action. In this work, we are examining skill, which could be understood from all (or any) parts of the video sequence. To make the most of the extent of the video sequence, we consider N uniform splits (Fig. 2a) and evaluate each of the corresponding splits in the loss function. We assume that two videos of the same task have comparable rate of progression through the task, and thus compare the temporal splits across a pair of videos in order. Assume p_i^k is the k^{th} split of video p_i, we extend the skill annotations such that,

$$E(p_i^k, p_j^k) = E(p_i, p_j) \quad \forall k = 1 \cdots N \tag{4}$$

Our loss function now becomes:

$$L_{rank2} = \sum_{(p_i,p_j)\in\Psi} \sum_{k=1}^{N} max(0, m - f_k(p_i) + f_k(p_j)) \tag{5}$$

In our experiments, $N = 7$ was tested. By pairing corresponding splits, we ensure the two videos are compared at a similar stage of the task performance, while still being able to deal with videos of different lengths, and therefore more discriminative features are likely to be learned.

3.6. Pairwise Deep Ranking with Similarity Loss

With the margin loss function in Section 3.4 we only incorporate pairs where one video is consistently ranked higher than another. In order to utilize more of the potential video pairings, we take inspiration from recent works in domain adaptation [10] by creating a secondary 'adversarial' loss where we wish to not distinguish between our similar pairs. We modify the margin loss to learn features which map pairs, indistinguishable in terms of skill, to similar scores. We thus find the set of pairs with indistinguishable skill levels $\Phi = \{(p_i, p_j); E(p_i, p_j) = 0\}$ (ref Eq. 1).

The way in which adversarial loss function are commonly created is by reversing the gradient, however this does not work in a ranking problem. In order to learn indistinguishable representations we aim for the following:

$$|f(p_i) - f(p_j)| \leq m \equiv |f(p_j) - f(p_i)| - m \leq 0 \tag{6}$$

Therefore, our new loss function for similar pairs becomes:

$$L_{sim} = \sum_{(p_i,p_j)\in\Phi} \sum_{k=1}^{N} max(0, |f(p_i) - f(p_j)| - m) \tag{7}$$

Resulting in a modified loss function:

$$L_{rank3} = \beta L_{rank2} + (1 - \beta) L_{sim} \tag{8}$$

Adding L_{sim} into our ranking loss for similarly ranked pairs not only allows us to utilize extra data pairs in the learning process, but also encourages the network to learn similarities in skill between similarly ranked videos. We explain how we get the new set of pairs Φ in Section 4.1.

3.7. Evaluating Skill for a Test Video

Following training, the learned 2S-CNN weights are used to evaluate the skill for test videos of the same task. In testing, we uniformly sample σ snippets from each video p_i, again as in [32]. Each snippet p_i^k $1 \leq k \leq \sigma$ is then fed into the spatial and temporal TSN independently. The output for each snippet is a score $f(p_i^k)$ for both spatial $f_s(p_i^k)$ and temporal $f_t(p_i^k)$ streams. To fuse the spatial and temporal networks for all snippets we take the weighted average of the outputs,

$$f(p_i) = \frac{1}{\sigma} \sum_{k=1}^{\sigma} \alpha f_s(p_i^k) + (1 - \alpha) f_t(p_i^k) \tag{9}$$

where α is the fusion weighting between spatial and temporal information, and σ is the number of testing snippets.

An overall ranking for a test set is achieved by ordering all test videos in a descending order based on $f(p_i)$.

4. Tasks and Datasets

For evaluation we conduct experiments on tasks from four datasets - two published and two newly recorded (Fig. 3). The first is a surgical dataset. Three other datasets containing daily living tasks are also used, to demonstrate the generality of the approach. Here we detail the four

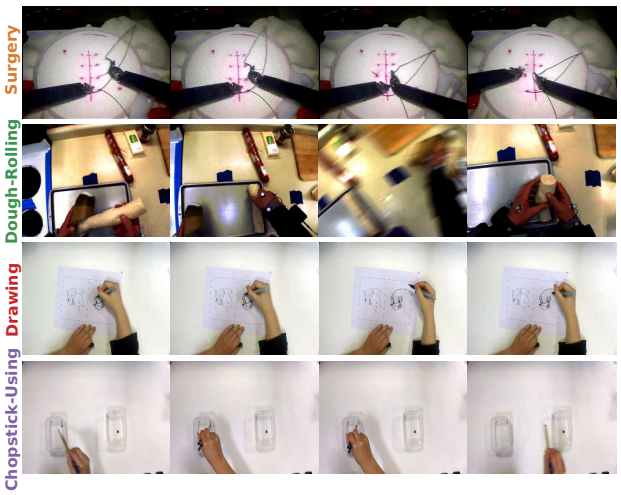

Figure 3. Sample sequences from the four tasks.

datasets, followed by the skill annotations for these datasets. These datasets and annotations will be combined to form the new EPIC-Skills 2018 dataset which can be found on the authors' webpages.

Surgery. We use the published JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS) dataset [11]. In this dataset, three surgical procedures are performed by 8 surgeons with varying levels of experience. In total, JIGSAWS consists of 36 trials of Knot Tying, 28 trials of Needle Passing and 39 trials of Suturing. This dataset contains stereo recordings, from which we use only one video (right view) from each sequence.

Dough-Rolling. We use the kitchen-based CMU-MMAC dataset [9], and select the dough rolling task from the pizza making activity, as this exhibits varying levels of performance across participants. In total, we manually segment 33 Dough-Rolling videos from 33 distinct participants.

Drawing. We introduce a new dataset for drawing, captured using a stationary camera at a resolution of 1920x1080 and a frame rate of 60 fps. Participants were given a reference image to copy. Two reference images were used; a cartoon of Sonic the Hedgehog and a gray-scale photograph of a hand. Similarly to the **Surgery** tasks, both tasks were performed five times each, by four participants.

Chopstick-Using. We also introduce a new dataset for using chopsticks, captured using the same setup as the **Drawing** dataset. Each participant was tasked with moving as many of the beans as possible from one tub to the other using chopsticks, limited to one minute per trial. Eight participants were recruited, each repeated the task five times.

4.1. Skill Annotation

Only the JIGSAWS dataset has existing skill scores. This was annotated by a surgery expert, out of a maximum score of 30. In this section, we explain how we obtained skill ranking for the remaining three datasets using *Amazon Mechanical Turk (AMT)*.

Task	#Videos	#Max Pairs	%Cons. Pairs	%Sim. Pairs	Total Pairs
Surgery (KT)	36	630	95%	5%	100%
Surgery (NP)	28	378	96%	4%	100%
Surgery (Suturing)	39	701	95%	5%	100%
Dough-Rolling	33	528	34%	18%	52%
Drawing (Sonic)	20	190	62%	37%	99%
Drawing (Hand)	20	190	68%	26%	94%
Chopstick-Using	40	780	69%	10%	79%

Table 1. For the four datasets: #videos, #of pairs $(n)(n-1)/2$ with the percentage of consistent pairs in annotations and similar pairs obtained. KT=Knot Tying, NP=Needle Passing

We determine the ground truth relative ranking of video pairs using a similar method to [21], where the authors demonstrate crowdsourcing yields reliable pairwise comparison for skill in surgical tasks. We asked AMT workers to watch pairs of videos simultaneously and select the video displaying the higher level of skill for the given task. Each worker was presented with 5 pairs of videos per HIT from the same task, one of which was a quality control pair which displayed an obvious difference in skill. Annotators were asked for strict preferences per pair. We then check for consensus between different annotators for skill annotation. Each video pair was annotated by four different workers. Only pairs of videos for which *all* annotators agreed on their skill order are considered for training in the L_{rank} loss function, we refer to these as *consistent pairs*.

We further check for any discrepancies in the set of these *consistent pairs*, by checking for *triangular inconsistencies*. Assume $E(p_i, p_j) > 0$ and $E(p_j, p_k) > 0$, we check for $E(p_i, p_k) < 0$, which would show a triangular inconsistency in annotations. We do this by creating a directed graph with P nodes and edges $(p_i \rightarrow p_j)\, \forall\, 1 \leq i, j \leq K$ where $E(p_i, p_j) > 0$. Cycles in the graph would indicate a triangular inconsistency, which we manually resolve. Only a single triangular inconsistency was found in all AMT annotations for the three tasks. This was in the **Dough-Rolling** task and was excluded from training and testing. Similarly, we take the skill scores from the **Surgery** dataset and compute all *consistent pairs*.

As well as the *consistent pairs* we use in L_{rank} we also require similarly ranked pairs for L_{sim} (Eq. 7). These are not all the *inconsistent pairs*, as those may be noisy. We select similar pairs for training using the directed graph of all pairs introduced above. We define separation between a pair of videos to be the difference in the length of the longest walk from any source node in the graph. We consider pairs in the set of *inconsistent* pairs with a separation of 0 or 1 as

Method	Surgery			Dough-Rolling			Drawing			Chopstick-Using		
	S	T	TS	S	T	TS	S	T	TS	S	T	TS
Siamese TSN with L_{rank1}	64.7	72.8	69.1	77.6	79.4	78.5	75.6	77.4	78.0	67.2	67.9	68.8
Siamese TSN with L_{rank2}	64.4	**73.3**	69.0	79.1	**80.4**	78.5	74.9	81.8	79.1	67.2	69.9	68.8
Siamese TSN with L_{rank3}	**66.4**	72.5	**70.2**	**79.5**	79.5	**79.4**	**77.6**	**82.7**	**83.2**	**70.8**	**70.6**	**71.5**

Table 2. Results of 4-fold cross validation on all datasets, for our proposed method with each of our proposed loss functions. For all datasets L_{rank3} outperforms original loss L_{rank1}. S=Spatial, T=Temporal, TS=Two-Stream

our set of *similar pairs*.

Table 1 presents statistics on these *consistent* and *similar* pairs. **Surgery** has a high number of consistent pairs ($> 95\%$). The pairs in this dataset come from the scores of a single expert, available with the JIGSAWS dataset, therefore pairs are only excluded when two videos have the same score. For the other tasks, we use the judgments from multiple AMT workers. **Dough-Rolling** has the lowest percentage of *consistent* pairs, as many were considered comparable in skill by human annotators. This is likely due to the nature of the task, thus many subjects do manage a similar level of performance. For **Drawing** and **Chopstick-Using**, the number of consistent pairs is $60 - 70\%$.

5. Experiments

For all datasets, we use a four-fold cross validation to report results. For each fold, the pairs between three quarters of the videos are used in training, and we then test on all remaining pairs. This includes pairs where neither video has been used in a pair for training as well as pairs where one video has been used in training within a different pairing.

5.1. Implementation Details

To extract the optical flow frames for the temporal network we use the $TV - L^1$ algorithm [34]. We use mini-batch stochastic gradient descent with a batch size of 128 and a momentum of 0.9. Both the spatial and temporal network use the AlexNet [19] architecture as we found this gave better results with a shorter training time than the BN-Inception [13] network original used in TSN. Both sides of each Siamese network are initialized with network weights from pre-trained ImageNet models. In the spatial network the learning rate begins as 1E-3 and decreases by a factor of 10 every 1.5K iterations, with the learning process finishing after 3.5K iterations. The temporal network's learning rate is initialized as 5E-3, decreasing by a factor of 10 after 10K iterations and after 16K iterations, with learning ending after 18K iterations. We set β (Eq. 8) to 0.5 in all experimental results after initial assessment.

To avoid over-fitting, we use the same data augmentation techniques as Wang *et al*. [32], namely horizontal flipping, corner cropping and scale jittering on the 340x256 pixels RGB and optical flow images. The cropped regions are 224x224 pixels for network training. We also use dropout layers with the fully connected layers, ratios used are 0.5 for both streams.

5.2. Evaluation Metric

To evaluate our method, we use pairwise precision on the rankings produced by each testing fold. Pairwise precision is defined as the *percentage of correctly ordered pairs in a ranking*. We say a pair is correctly ordered if for a pair (p_i, p_j) where $E(p_i, p_j) = 1$ in the ground truth, the method outputs $f(p_i) > f(p_j)$.

5.3. Results

In Table 2 we show our results from four-fold cross-validation on each of the four datasets with each loss function. We report results with $\sigma = 25$ as in [32], and for $\alpha = 0.4$ (Eq. 9) as in [29, 32]. Below we test the sensitivity of these results to the values of α. From this Table 2, we can see that our proposed loss function L_{rank3} outperforms the standard margin loss function L_{rank1} on all combinations of modality and dataset. We also see an improvement from L_{rank2} over all but the temporal result in Surgery and Dough-Rolling. This improvement is particularly noticeable in the two-stream results for Drawing (79.1% to 83.2%) and Chopstick-Using (68.8% to 71.5%). The inclusion of similar pairs with L_{sim} in the training process has the largest impact on training on the spatial network, where the results for skill determination are generally weaker across tasks. L_{sim} increases the spatial network performance resulting in larger improvements in the two-stream result.

From the results in Table 2 we can also conclude that the temporal features are in general more useful for determining skill for the presented tasks, with the temporal result outperforming the spatial result in all but the Chopstick-Using task. Although, we do manage to reduce this gap with our additional loss L_{sim} from Section 3.4. This implies the motions performed are more important for determining skill than the current state or appearance of the task (captured in the spatial stream). We note that the largest difference between the two streams is in the Surgery tasks. This is

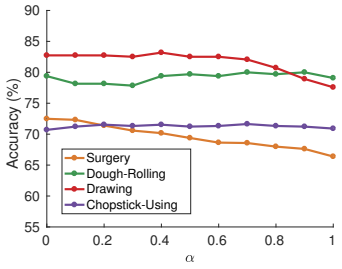

Figure 4. The accuracy for each dataset with different α values. The method is resilient to the parameter value chosen.

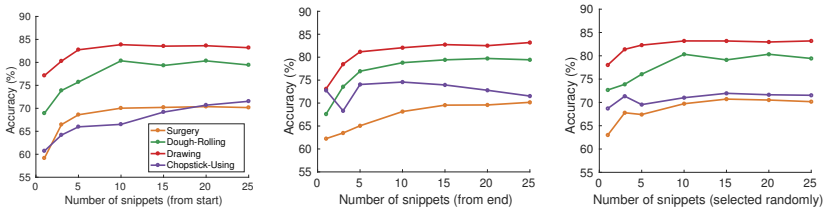

Figure 5. The accuracy achieved when adding snippets in testing, from the start (left), end (middle) and randomly (right).

because these tasks require quick smooth motions, putting minimal stress on the surrounding areas. Hence, while the end result of each stage is visually similar, the motions affect the scoring significantly.

Fusion Parameter. We assess the sensitivity of our results to the late fusion weighting α in Equation 9. We test α values from 0 to 1 at intervals of 0.1 for all datasets, as shown in Figure 4. For the majority of tasks, the combination of temporal and spatial modalities is useful, except the Surgery task which peaks at $\alpha = 0$, i.e. no information from the spatial network is included. All tasks benefit from the contribution of the temporal network, albeit to different degrees. We also observe that the method is highly resilient to the value of α, particularly for the Chopstick-Using and Dough-Rolling tasks.

Number of Snippets in Testing. The results above are reported for $\sigma = 25$ (in-line with previous methods), where the snippets are sampled uniformly from the whole video. However, it is interesting to examine how much of the video is needed at test time to gain an accurate evaluation of the skill displayed in a video. We test our evaluation on a varying number of consecutive snippets from the start and end of the video, as well as randomly. Results are shown in Fig. 5.

From Figure 5 we can see that good accuracy can be obtained after only seeing a portion of the video, however a single frame of the video is insufficient to measure skill, even if this single frame contains the end result of the task, and accuracy improves as further snippets are viewed. Interestingly, accuracy converges as the number of snippets continues to increase, for the various tasks and snippet sampling approaches. For instance, the Surgery task achieves near peak accuracy with the first 20% of the testing snippets, while the last 20% appear redundant. This difference is intuitive as the start of the Surgery task is more challenging, while the repetitive nature of the task allows novice participants to improve by the end.

5.4. Baselines

As there are no generic existing methods for ranking skill nor performing skill determination for non-surgical tasks, we use existing ranking methods developed for other applications. Our first baseline uses RankSVM [15], commonly used in ranking problems [22]. We perform four-fold cross validation on RankSVM with features extracted from AlexNet trained on ImageNet [19] and C3D trained on Sports1M [30]. These two results are then combined with late fusion of $\alpha = 0.4$. We use the same implementation as in [16] which can be found on the authors webpage.

Method	Surgery	Dough-Rolling	Drawing	Chopstick-Using
RankSVM [16]	65.2	72.0	71.5	**76.6**
Yao *et al*. [33]	66.1	78.1	72.0	70.3
Ours	**70.2**	**79.4**	**83.2**	71.5

Table 3. Results of 4-fold cross validation on all datasets, for the baselines our proposed method with L_{rank3}.

The second baseline is Yao *et al*. [33] who originally performed deep ranking on video to determine highlight segments. They use pre-extracted features in a fully connected network to determine segments with the highest potential highlight score. The features used are from AlexNet and C3D which are then passed separately into a network with architecture: $F1000$-$F512$-$F256$-$F128$-$F64$-$F1$ using the same margin loss in Eq. 3. The results from each network are then fused using late fusion with $\alpha = 0.4$. Although this method was originally developed for a different purpose - binary highlight detection - it uses a general method of ranking video and is thus used here for comparison.

It is important to note, the only dataset for which skill evaluation has previously been considered is the Surgery dataset, though as a regression problem to expert scores. This approach is not applicable to daily tasks where obtaining objective scoring is much harder than pairwise ranking. Published results on the Surgery dataset either report Expert/Novice classification [1] or use only the kinematic data [38] and are therefore not comparable to ours.

Comparative results are available in Table 3. Our method outperforms both baselines on three of the four tasks.

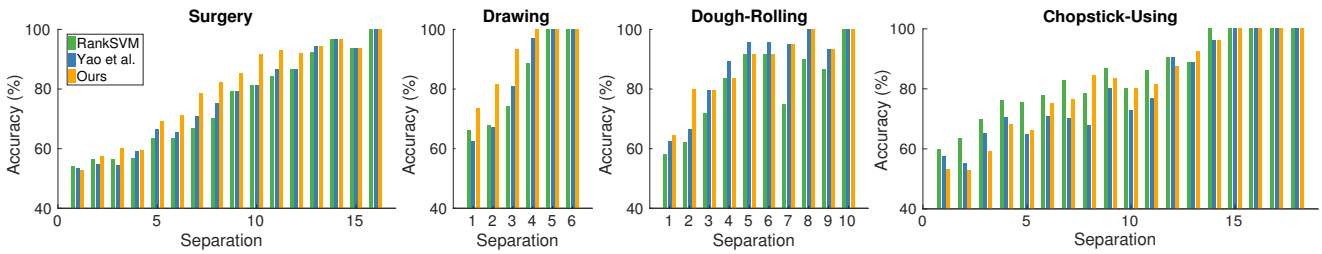

Figure 6. The accuracy of each ordered pair by separation between videos in a pair for each task, in the baselines and our method. The accuracy consistently increases as tested pairs are further in the ground-truth ranking for all datasets.

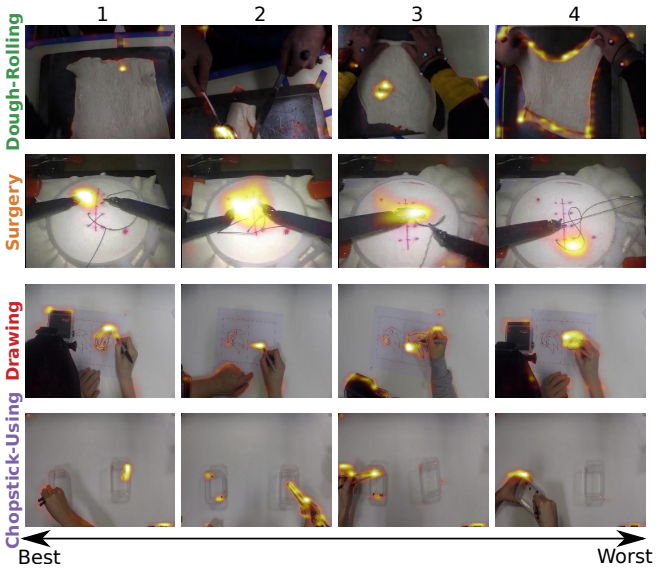

Figure 7. Spatial activations for sample frames at varying ranks.

RankSVM performs best on Chopstick-Using. The improvement with our method is most significant in the Drawing task with an improvement of 11.2%.

To study where the difference in performance lies, we show the accuracy for each level of separation between pairs of each dataset in Figure 6. Assume we have consistent annotation pairings resulting in the partial ranking $p_i < p_{i+1} < ... < p_{i+n} < p_j$, then we define separation between p_i and p_j as $n + 1$. It is more important that pairs with high separation be correctly ordered than pairs close together in the ranking. Figure 6 shows that the significant improvement of our method, over the baselines, in Surgery and Drawing is at the mid-level of separation. Although all methods approach the 100% accuracy for the most separated pairs, our method approaches this much faster. Alternatively, in the Chopstick-Using task, the only task for which we perform below one baseline, we have comparative performance in the mid and high separation compared to RankSVM, only falling below for nearby pairs.

5.5. Visualizing Performance Ranking

A key difficultly of skill determination is capturing the nuance of the tasks in the learned model. In Fig. 7 we visualize the top-down attention of the spatial CNN on example rankings for three datasets using [26] based on [35]. For each dataset, we show frame-level spatial activations on four videos with varying levels of skill (best→worst).

From Fig.7 we can see that the trained model is picking details that correspond to what a human would attend to. In Dough-Rolling high activations occur on holes in the dough (1, 3), curved or rolled edges (4) and when using a spoon (2). High activations occur in Surgery when strain is put on the material (1, 2), with abnormal needle passes (3) and when there is loose stitching (4). In Drawing, the model attends to specific parts of the sketch such as the head and mouth. The high activations in the Chopstick-Using task occur on the hand position (3,4), chopstick position (2) and the bean locations (1,2,3). Further qualitative results are shown in the supplementary video.

6. Conclusion

In this paper we have presented a method to rank videos based on the skill that subjects demonstrate. Particularly, we have proposed a pairwise deep ranking model which utilizes both spatial and temporal streams in combination with a novel loss to determine and rank skill. We have tested this method on four separate datasets, two newly created, and show that our method outperforms the baseline on three out of four datasets, with all tasks achieving over 70% accuracy. Furthermore, we have explored where the performance increase lies and examined our method's resistance to changes in parameters. Qualitative figures demonstrate the approach's ability to learn tasks' nuances, while using a general, task-independent, method.

We see our work as a promising step toward the automated and objective organization of *how-to* video collections and as a framework to motivate more work in skill determination from video. Further work involves exploring mid-level fusion between the two streams of the network, as well as testing on additional and across datasets and tasks.

Acknowledgements: Access to EPIC-Skills 2018 dataset and annotations available from authors' webpages. Supported by an EPSRC DTP and EPSRC GLANCE (EP/N013964/1).

References

[1] N. Ahmidi, L. Tao, S. Sefati, Y. Gao, C. Lea, B. Bejar, L. Zappella, S. Khudanpur, R. Vidal, and G. D. Hager. A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery. *IEEE Transactions on Biomedical Engineering*, 2017. 7

[2] J.-B. Alayrac, P. Bojanowski, N. Agrawal, I. Laptev, J. Sivic, and S. Lacoste-Julien. Unsupervised learning from narrated instruction videos. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[3] J.-B. Alayrac, I. Laptev, J. Sivic, and S. Lacoste-Julien. Joint discovery of object states and manipulation actions. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[4] G. Bertasius, H. Soo Park, S. X. Yu, and J. Shi. Am i a baller? basketball performance assessment from first-person videos. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[5] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96. ACM, 2005. 2

[6] C. J. Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81, 2010. 2

[7] O. Çeliktutan, C. B. Akgul, C. Wolf, and B. Sankur. Graph-based analysis of physical exercise actions. In *Proceedings of the 1st ACM international workshop on Multimedia indexing and information retrieval for healthcare*, pages 23–32. ACM, 2013. 2

[8] D. Damen, T. Leelasawassuk, and W. Mayol-Cuevas. You-Do, I-Learn: Egocentric unsupervised discovery of objects and their modes of interaction towards video-based guidance. *Computer Vision and Image Understanding (CVIU)*, 2016. 2

[9] F. De la Torre, J. Hodgins, A. Bargteil, X. Martin, J. Macey, A. Collado, and P. Beltran. Guide to the carnegie mellon university multimodal activity (CMU-MMAC) database. *Robotics Institute*, page 135, 2008. 5

[10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. 4

[11] Y. Gao, S. Vedula, C. Reiley, N. Ahmidi, B. Varadarajan, H. Lin, L. Tao, L. Zappella, B. Béjar, D. Yuh, et al. The JHU-ISI gesture and skill assessment dataset (JIGSAWS): A surgical activity working set for human motion modeling. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2014. 2, 3, 5

[12] W. Ilg, J. Mezger, and M. Giese. Estimation of skill levels in sports based on hierarchical spatio-temporal correspondences. In *Joint Pattern Recognition Symposium*, pages 523–531. Springer, 2003. 2

[13] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 448–456, 2015. 6

[14] K. Järvelin and J. Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41–48. ACM, 2000. 2

[15] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM, 2002. 2, 7

[16] T. Joachims. Training linear svms in linear rime. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226. ACM, 2006. 7

[17] M. Jug, J. Perš, B. Dežman, and S. Kovačič. Trajectory based assessment of coordinated human activity. In *International Conference on Computer Vision Systems*, pages 534–543. Springer, 2003. 2

[18] S. T. Kim and Y. M. Ro. Evaluationnet: Can human skill be evaluated by deep networks? *arXiv preprint arXiv:1705.11077*, 2017. 2

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 6, 7

[20] A. Malpani, S. S. Vedula, C. C. G. Chen, and G. D. Hager. Pairwise comparison-based objective score for automated skill assessment of segments in a surgical task. In *International Conference on Information Processing in Computer-Assisted Interventions*, pages 138–147. Springer, 2014. 2

[21] A. Malpani, S. S. Vedula, C. C. G. Chen, and G. D. Hager. A study of crowdsourced segment-level surgical skill assessment using pairwise rankings. *International journal of computer assisted radiology and surgery*, 10(9):1435–1447, 2015. 5

[22] D. Parikh and K. Grauman. Relative attributes. In *IEEE International Conference on Computer Vision (ICCV)*, pages 503–510, 2011. 2, 7

[23] G. I. Parisi, S. Magg, and S. Wermter. Human motion assessment in real time using recurrent self-organization. In *Robot and Human Interactive Communication (RO-MAN), 2016 25th IEEE International Symposium on*, pages 71–76. IEEE, 2016. 2

[24] P. Parmar and B. T. Morris. Learning to score olympic events. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 76–84. IEEE, 2017. 2

[25] H. Pirsiavash, C. Vondrick, and A. Torralba. Assessing the quality of actions. In *European Conference on Computer Vision (ECCV)*, pages 556–571. Springer, 2014. 2

[26] W. Price. Two stream action CNN analysis - code. https://github.com/willprice/two-stream-action-cnn-analysis/, 2017. 8

[27] Y. Sharma, V. Bettadapura, T. Plötz, N. Hammerla, S. Mellor, R. McNaney, P. Olivier, S. Deshmukh, A. McCaskie, and I. Essa. Video based assessment of osats using sequential motion textures. In *International workshop on modeling and monitoring of computer assisted interventions (M2CAI)-workshop*, 2014. 2

[28] Y. Sharma, T. Plötz, N. Hammerld, S. Mellor, R. McNaney, P. Olivier, S. Deshmukh, A. McCaskie, and I. Essa. Automated surgical osats prediction from videos. In *Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on*, pages 461–464. IEEE, 2014. 2

[29] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems (NIPS)*, pages 568–576, 2014. 4, 6

[30] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497, 2015. 7

[31] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1386–1393, 2014. 4

[32] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision (ECCV)*, pages 20–36. Springer, 2016. 3, 4, 6

[33] T. Yao, T. Mei, and Y. Rui. Highlight detection with pairwise deep ranking for first-person video summarization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 982–990, 2016. 2, 4, 7

[34] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint Pattern Recognition Symposium*, pages 214–223. Springer, 2007. 6

[35] J. Zhang, Z. Lin, S. X. Brandt, Jonathan, and S. Sclaroff. Top-down neural attention by excitation backprop. In *European Conference on Computer Vision (ECCV)*, 2016. 8

[36] Q. Zhang and B. Li. Video-based motion expertise analysis in simulation-based surgical training using hierarchical dirichlet process hidden markov model. In *Proceedings of the 2011 international ACM workshop on Medical multimedia analysis and retrieval*, pages 19–24. ACM, 2011. 2

[37] Q. Zhang and B. Li. Relative hidden markov models for video-based evaluation of motion skills in surgical training. *IEEE transactions on pattern analysis and machine intelligence*, 37(6):1206–1218, 2015. 2

[38] A. Zia and I. Essa. Automated surgical skill assessment in rmis training. *arXiv preprint arXiv:1712.08604*, 2017. 7

[39] A. Zia, Y. Sharma, V. Bettadapura, E. Sarin, T. Ploetz, M. Clements, and I. Essa. Automated video-based assessment of surgical skills for training and evaluation in medical schools. *International journal of computer assisted radiology and surgery*, 11(9):1623, 2016. 2

[40] A. Zia, Y. Sharma, V. Bettadapura, E. L. Sarin, M. A. Clements, and I. Essa. Automated assessment of surgical skills using frequency analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 430–438. Springer, 2015. 2

[41] A. Zia, Y. Sharma, V. Bettadapura, E. L. Sarin, and I. Essa. Video and accelerometer-based motion analysis for automated surgical skills assessment. *International journal of computer assisted radiology and surgery*, 13(3):443–455, 2018. 2