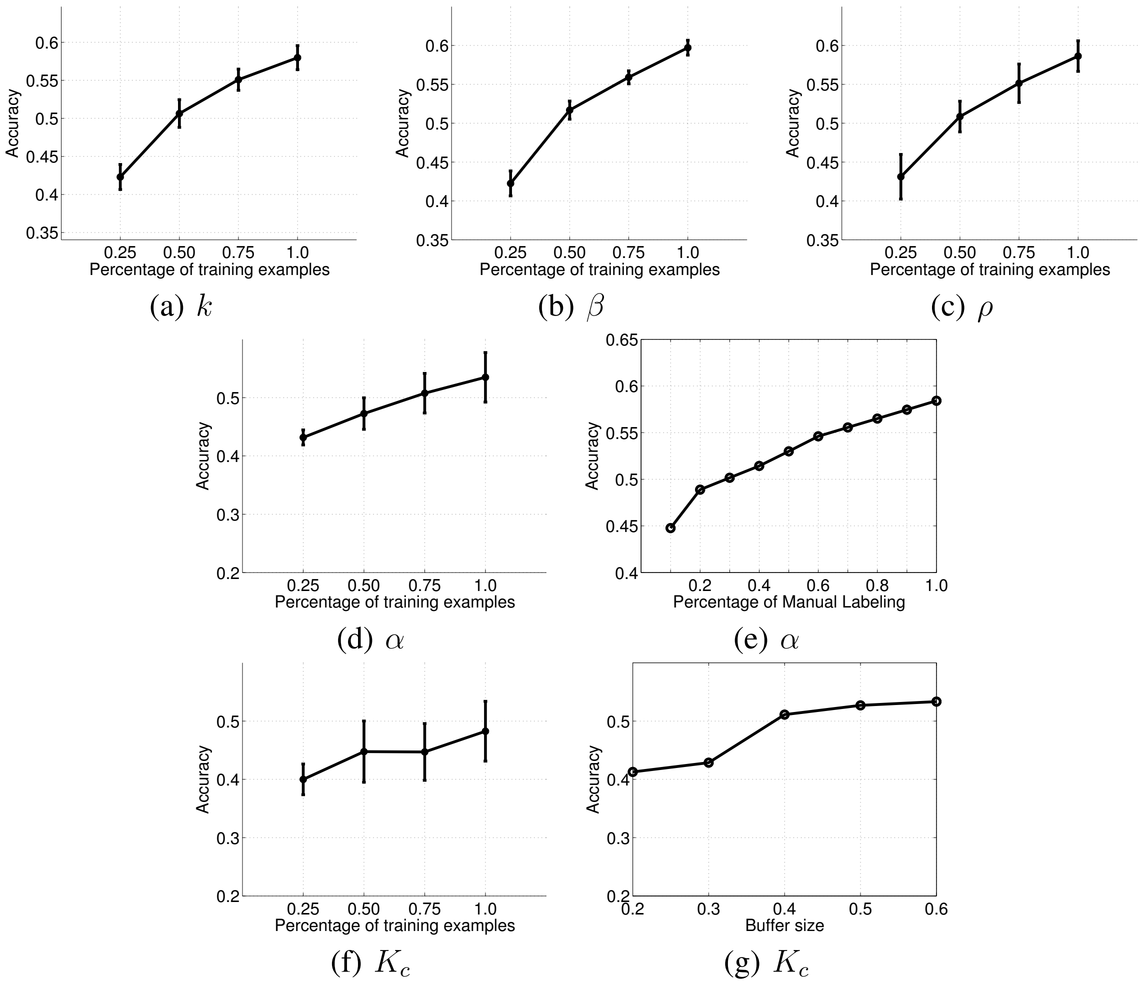
Continuous Learning of Human Activity Models using Deep Nets

Supplementary Materials

TABLE I
TABLE OF CONTENTS.

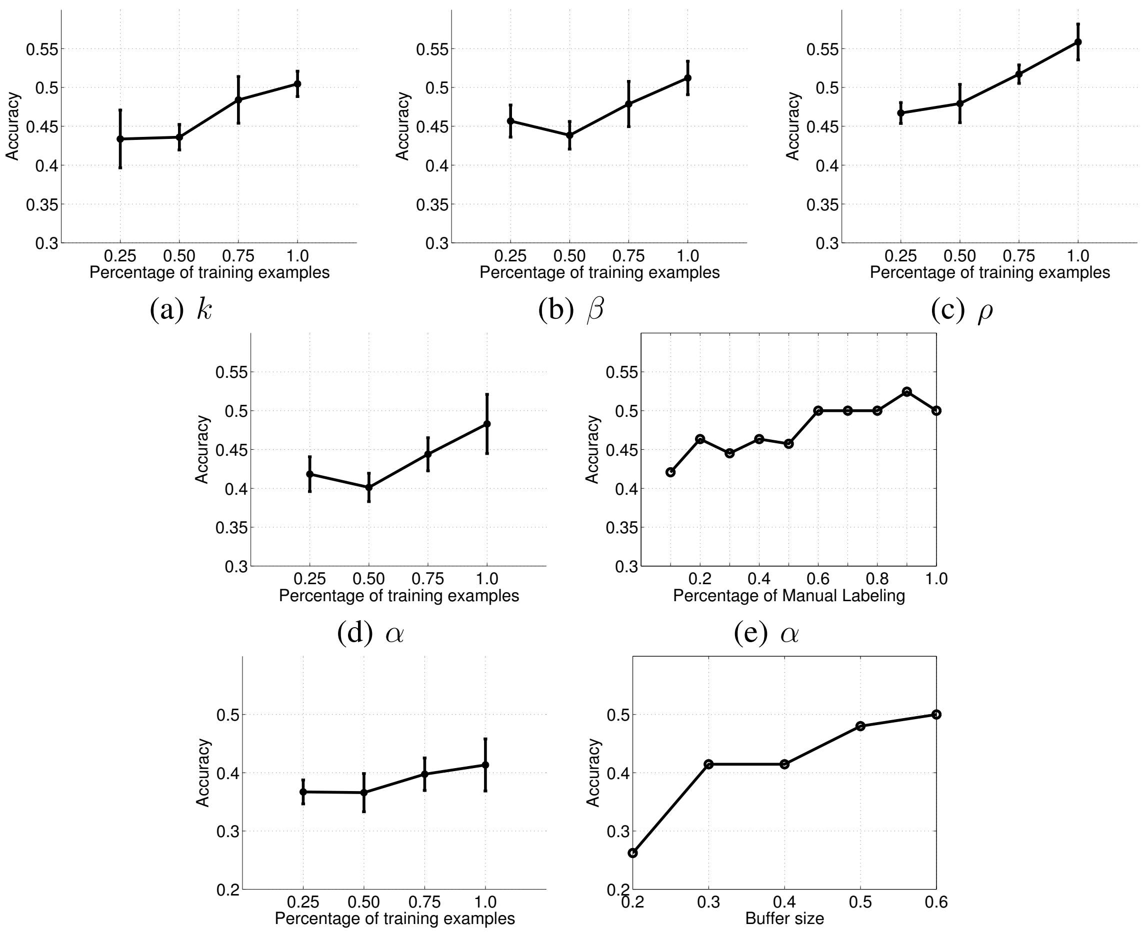
UCF11 Parameter Sensitivity



(a) k (b) β (c) ρ (d) α (e) α (f) K_c (g) K_c

Fig. 1. (a) Number of neurons in the hidden layer k vary from 100 to 1500 with 100 increment. (b) β vary from 0.5 to 5 with 0.5 increment (c) ρ vary from 0.05 to 0.5 with 0.05 increment. Performance variation is not significant with the change of k and β, while for ρ performance variation is relatively more significant. (d) and (e) show the effect of the amount of manual labeling. Performance variation is large as expected. As more instances are presented to the framework performance variation increases significantly (plot (d)). If we increase the amount of manual labeling by active learning (changing α), performance increases as shown in plot (e). However, performance with 60%-70% manual labeling is closer to the performance with 100% manual labeling. (f) and (g) show the effect of buffer size K_c, which has significant effect on the performance. Performance increases with buffer size as expected. In plot (g), x-axis shows the buffer size as the fraction of total number of training instances available. It is evident that, we can achieve satisfactory performance by storing less number of instances in the memory.

VIRAT Parameter Sensitivity

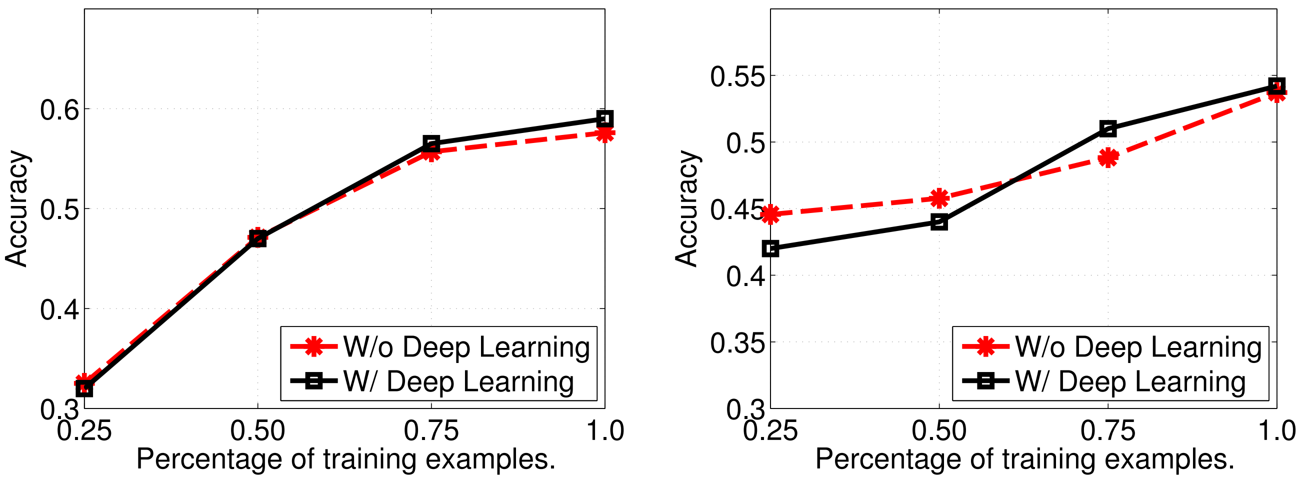


(a) k (b) β (c) ρ

(d) α (e) α

(f) K_c (g) K_c

Fig. 2. (a) Number of neurons in the hidden layer k vary from 100 to 1500 with 100 increment. (b) β vary from 0.5 to 5 with 0.5 increment (c) ρ vary from 0.05 to 0.5 with 0.05 increment. Performance variation is not significant with the change of ρ, while for k and β performance variation is relatively more significant. (d) and (e) show the effect of the amount of manual labeling. Performance variation is large as expected. However, it is interesting that with around 50%-60% manual labeling our framework can achieve performance close to 100% manual labeling. (f) and (g) show the effect of buffer size K_c, which has significant effect on the performance. Performance increases with buffer size as expected. In plot (g), x-axis shows the buffer size as the fraction of total number of training instances available. It is evident that, we can achieve satisfactory performance by storing less number of instances in the memory.



(a) (b)

Fig. 3. Plot (a) shows the benefit of using deep learning on UCF11 dataset. The activity model that does not use deep learning is outperformed by the method (A0F0) that use deep learning by a margin of around 1%. Plot (b) shows the benefit of using deep learning on VIRAT dataset. Performance of the activity model that does not use deep learning is better initially, but as more instances are presented to the framework deep learning based method (A0F0) outperform other method by a margin of 0.5%. It demonstrates the ability of learning of our framework in concept drift.

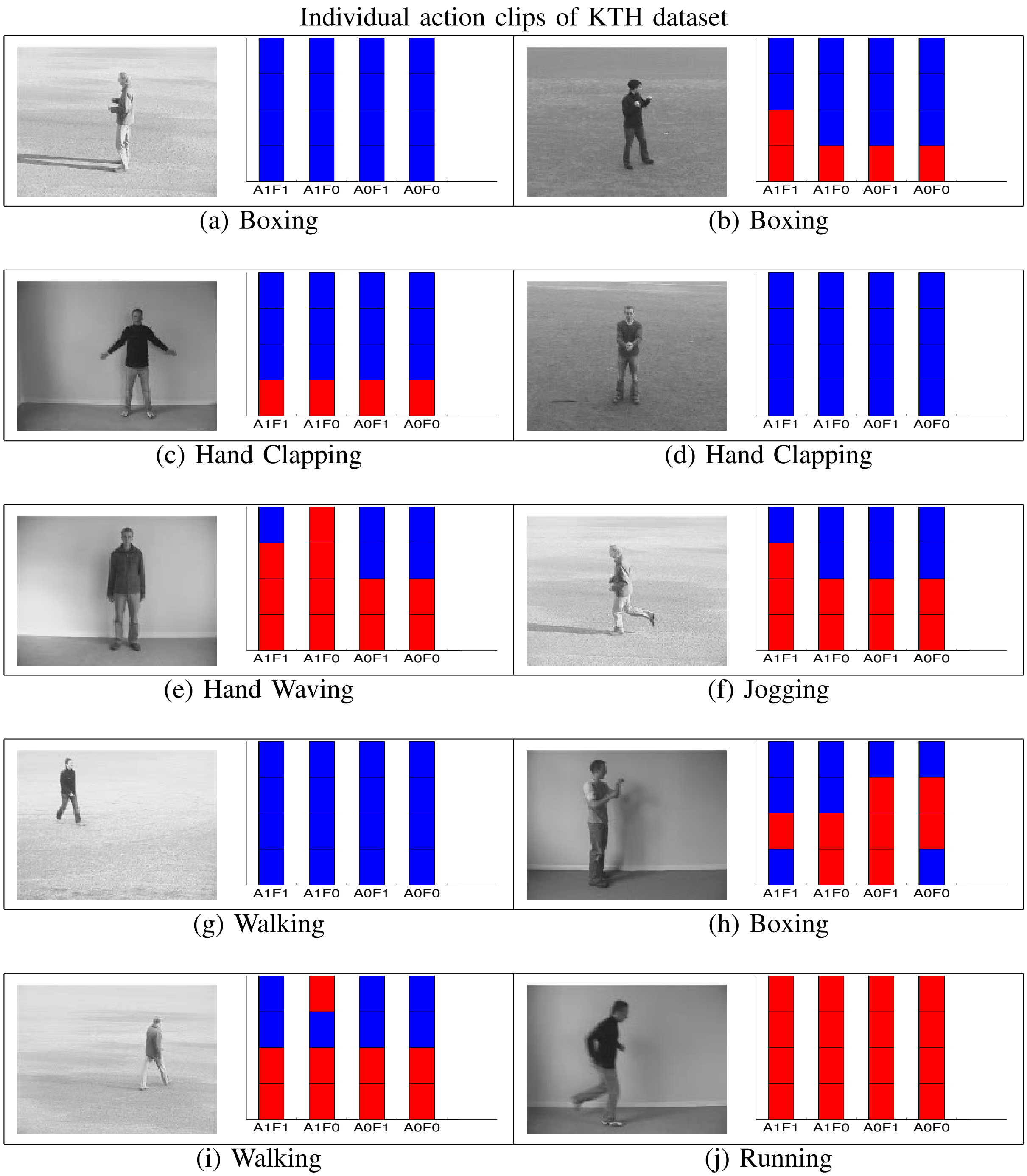


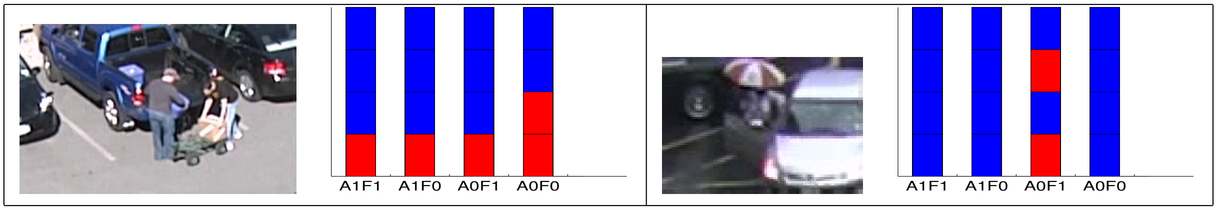
Fig. 4. Evaluation of continuous leanring on individual action clips of KTH dataset. Each row of this figure has two boxes and each box contains a snap of an action clip to the left and its corresponding evaluaiton to the right (as stacked bar chart). The bar chart has four bars for four methods such as A1F1, A1F0, A0F1, and A0F0 respectively from left to right. Each bar has four segments illustraing four batches of continuous learning with 25%, 50%, 75%, and 100% training instances presented so far to the framework respectively from bottom to top. Each segment is colored either blue or red. A blue segment means that the action clip is classified correctly after that batch of training instacnes are presented and red means misclassificaiton. For some cases such as (a), (d), and (g), the action clip is correctly classified from first batch through last batch. For some cases such as (b), (c), (e), and (f), the action clip is missclassified initially but correctly classified later as more training instances are presented to the framework. For some cases such as (h) and (i), classification decision may change intermittently. For few hard instances such as (j), action clip remains missclassfied after all batches of training.

Individual action clips of UCF11 dataset

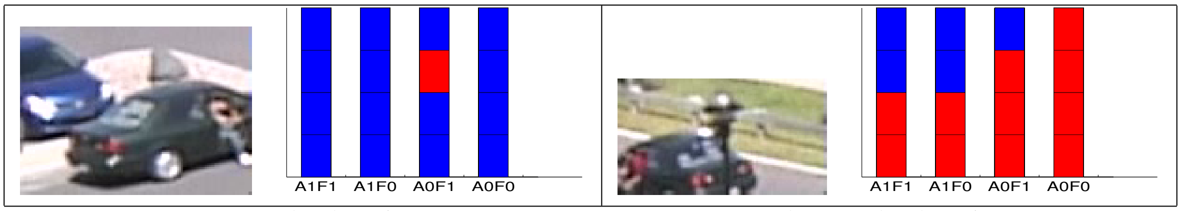
(a) Biking (b) Basketball (c) Diving (d) Golf Swing (e) Soccer Juggling (f) Swing (g) Tennis Swing (h) Jumping (i) Walking (j) Spiking

Fig. 5. Evaluation of continuous leanring on individual action clips of UCF11 dataset. Each row of this figure has two boxes and each box contains a snap of an action clip to the left and its corresponding evaluaiton to the right (as stacked bar chart). The bar chart has four bars for four methods such as A1F1, A1F0, A0F1, and A0F0 respectively from left to right. Each bar has four segments illustraing four batches of continuous learning with 25%, 50%, 75%, and 100% training instances presented so far to the framework respectively from bottom to top. Each segment is colored either blue or red. A blue segment means that the action clip is classified correctly after that batch of training instacnes are presented and red means misclassificaiton. For some cases such as (a), and (d), the action clip is correctly classified from first batch through last batch. For some cases such as (b), (c), (e), (f), (g), (h), and (i), the action clip is missclassified initially but correctly classified later as more training instances are presented to the framework. For some cases such as (e), classification decision may change intermittently. For few hard instances such as (j), action clip remains missclassfied after all batches of training.

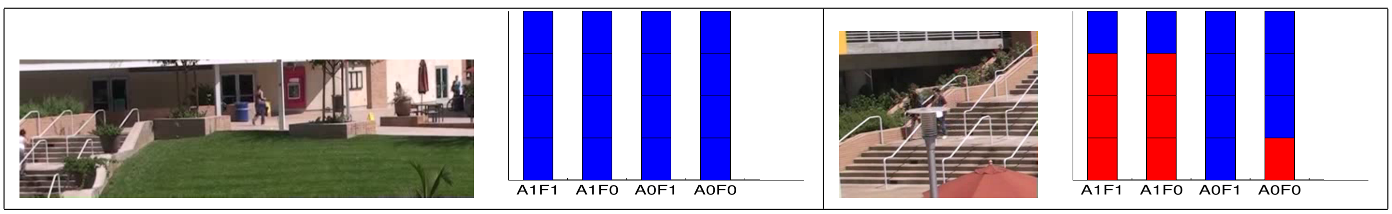
Individual action clips of VIRAT dataset



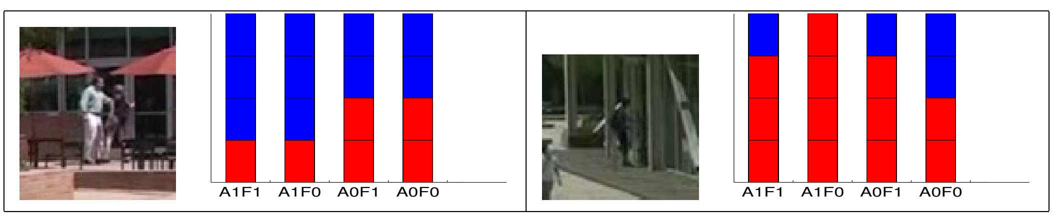
(a) Boxing (b) Boxing



(c) Hand Clapping (d) Hand Clapping



(e) Hand Waving (f) Jogging



(g) Walking (h) Boxing

(i) Walking (j) Running

Fig. 6. Evaluation of continuous leanring on individual action clips of VIRAT dataset. Each row of this figure has two boxes and each box contains a snap of an action clip to the left and its corresponding evaluaiton to the right (as stacked bar chart). The bar chart has four bars for four methods such as A1F1, A1F0, A0F1, and A0F0 respectively from left to right. Each bar has four segments illustraing four batches of continuous learning with 25%, 50%, 75%, and 100% training instances presented so far to the framework respectively from bottom to top. Each segment is colored either blue or red. A blue segment means that the action clip is classified correctly after that batch of training instacnes are presented and red means misclassificaiton. For some cases such as (e), and (i), the action clip is correctly classified from first batch through last batch. For some cases such as (a), (d), (f), (g), (h), and (j), the action clip is missclassified initially but correctly classified later as more training instances are presented to the framework. For some cases such as (b), and (c), classification decision may change intermittently.

Parameter Values and Sensitivity: We have three types of parameters, newly feature selection (T,L, and k), model training (β, rho, and λ), and experiment design parameters (K_c and α). Experimental result illustrated in Fig. 7 of the main paper was conducted using the empirically estimated parameter values shown in Table II. Choice of some of these parameter values has relatively significant effect on the performance as shown in the Fig. 8 of the main paper and Fig. 1 and 2 of this supplementary respectively for KTH, UCF11, and VIRAT datasets. However, most of the parameter values are same for different datasets.

	Feature				Model training			Design	
Dataset	**T**	**L**	**n**	**k**	λ	β	ρ	$\mathbf{K_c}$	α
KTH	1	3	2268	400	$10e^{-4}$	2	0.3	0.4	0.5
UCF11	2	3	4536	800	$10e^{-4}$	3.5	0.3	0.5	0.5
VIRAT	2	3	4536	800	$100e^{-4}$	2	0.1	0.5	0.5
TRECVID	2	3	4536	800	$10e^{-4}$	3	0.3	0.5	0.5

TABLE II
PARAMETER VALUES