Beyond Comparing Image Pairs: Setwise Active Learning for Relative Attributes

Supplementary Material

Lucy Liang and Kristen Grauman
The University of Texas at Austin
lliang@cs.utexas.edu, grauman@cs.utexas.edu

This file contains pseudo-code for the DSLM search procedure (Sec. 1), a preliminary test for testing robustness to K (Sec. 2), an explanation of how we computed annotation cost savings (Sec. 3), and per-attribute learning curves for all datasets (Sec. 4).

Note that Figure 7 in the main paper contains curves summarizing results over ALL attributes on all datasets. There are no new results in this document.

1. Pseudo-code

Algorithm 1 gives pseudo-code for the search procedure we devised for the diverse setwise margin criterion (Section 3.2.3 in the main paper). Here i_{vec} denotes a candidate set of k indices within the pool of images $\mathcal{P}$. $M_{i_{vec}}$ denotes the total margin, that is, the sum of distances between each data pair in i_{vec}. As explained in the paper, the "worst" offender is the one whose total margin distance to the other instances in i_{vec} is the largest. The diversity condition checks whether each member in the set i_{vec} originates from a different cluster. The method returns in I_{vec} the best k-set found.

As discussed in the main paper, only a rank-contiguous set of points can minimize the total margin (ignoring diversity constraints). There are two possible scenarios. In the first scenario, there exists a strictly rank-contiguous set with minimum total margin distances that is also diverse. In this case, since our optimization procedure visits each rank-contiguous set, we are guaranteed to return that optimal set. Its total margin score M_{curr} will be lower than any others found in the search. In the second scenario, there does not exist a rank-contiguous set that is also diverse. In this case, our approach perturbs the members of a candidate set (i.e., a k-set with the lowest total margin observed so far) until it is diverse. Each perturbation consists of an incremental swap, dropping one point and adding another. Namely, we add the point ranked just higher than the maximally ranked point in the set, and drop the one whose summed margin distances to the incremented set is maximal. Note that this swap is guaranteed to minimally increase the total mutual margin for the set, since the points are sorted by their (1D) predicted ranks. Our sub-routine performs a series of such swaps until the set is diverse, or until no higher-ranked points j exist (as seen in the "while" loop in Algorithm 1).

Algorithm 1 Diverse setwise low-margin selection

```
I_vec ⇐ ∅, M_min ⇐ inf                                  ▷ initialization
for i = 1 → |P| − k do
    i_vec ⇐ {i, i + 1, ..., i + k − 1}
    j ⇐ i + k
    M_curr ⇐ M_{i_vec}
    while UPDATE(M_min, M_curr, I_vec, i_vec) = false ∧ j ≤ |P| do
        o ⇐ index of worst "offender" in i_vec.
        i_vec ⇐ i_vec − {o}
        i_vec ⇐ i_vec ∪ {j}
        M_curr ⇐ M_{i_vec}
        j ⇐ j + 1
    end while
return I_vec

function UPDATE(M_min, M_curr, I_vec, i_vec)
    if Diversity Condition holds in i_vec then
        M_min ⇐ M_curr
        I_vec ⇐ i_vec
        return true
    else
        return false
```

2. Robustness to K selection

Figure 1 shows a preliminary test indicating the insensitivity of results with respect to values of K in the vicinity of the number of object classes in the dataset, which was our prior for fixing K. See Lines 551-557 and 636-638 in the main text. Notice there is little difference in performance for values $5 < K < 20$.

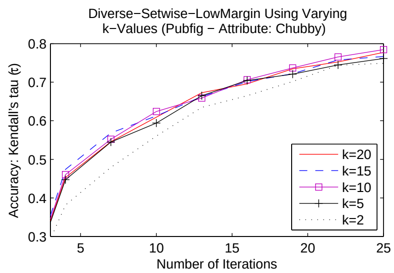

Figure 1: Learning curves for DSLM with varying K values used to enforce diversity.

3. Annotation effort saved

Our estimate of saving 39% in total annotation effort (Line 734 in the main text) is based on the following analysis. We multiply the number of total annotation iterations by the number of attributes, per dataset. This yields $13 \times 20 = 260$ (Shoes), $13 \times 11 = 143$ (PubFig), and $7 \times 6 = 42$ (OSR) rounds of annotation, for a total of $42 + 260 + 143 = 445$ total annotations. After applying all the annotations, the passive learner achieves a certain rate of accuracy. Now, we see at which iteration (per attribute and dataset) our method achieves the same level of accuracy, and tally those values, arriving at 274 iterations by our method. So, our method uses 274/445 = 61% of the annotations, or saves 39% in annotations.

As explained in the text (see Lines 514-523 and 643-661), we set k to equalize the effort required for partial orders or pairs. So, the number of iterations represents annotation cost.

4. Per attribute learning curves

Figure 7 in the main paper contains curves summarizing results over ALL attributes on all datasets, since we did not have space to plot them separately. We include the individual attribute results below for completeness. We stress, however, that they do not introduce any new findings not already presented in the main paper.

For the offline experiments, Figures 2, 3, and 4 show the results for all attributes in the Shoes, OSR, and PubFig datasets, respectively. We report averages and standard errors over 20 trials per attribute.

For the live experiments, Figures 5, 6, and 7 show the results for all attributes in the Shoes, OSR, and PubFig datasets, respectively. The individual curves are more "bumpy" than those for the offline experiments. This is to be expected, since the ground truth for the offline experiments permits averaging over 20 trials.

Please see the main paper for discussion.

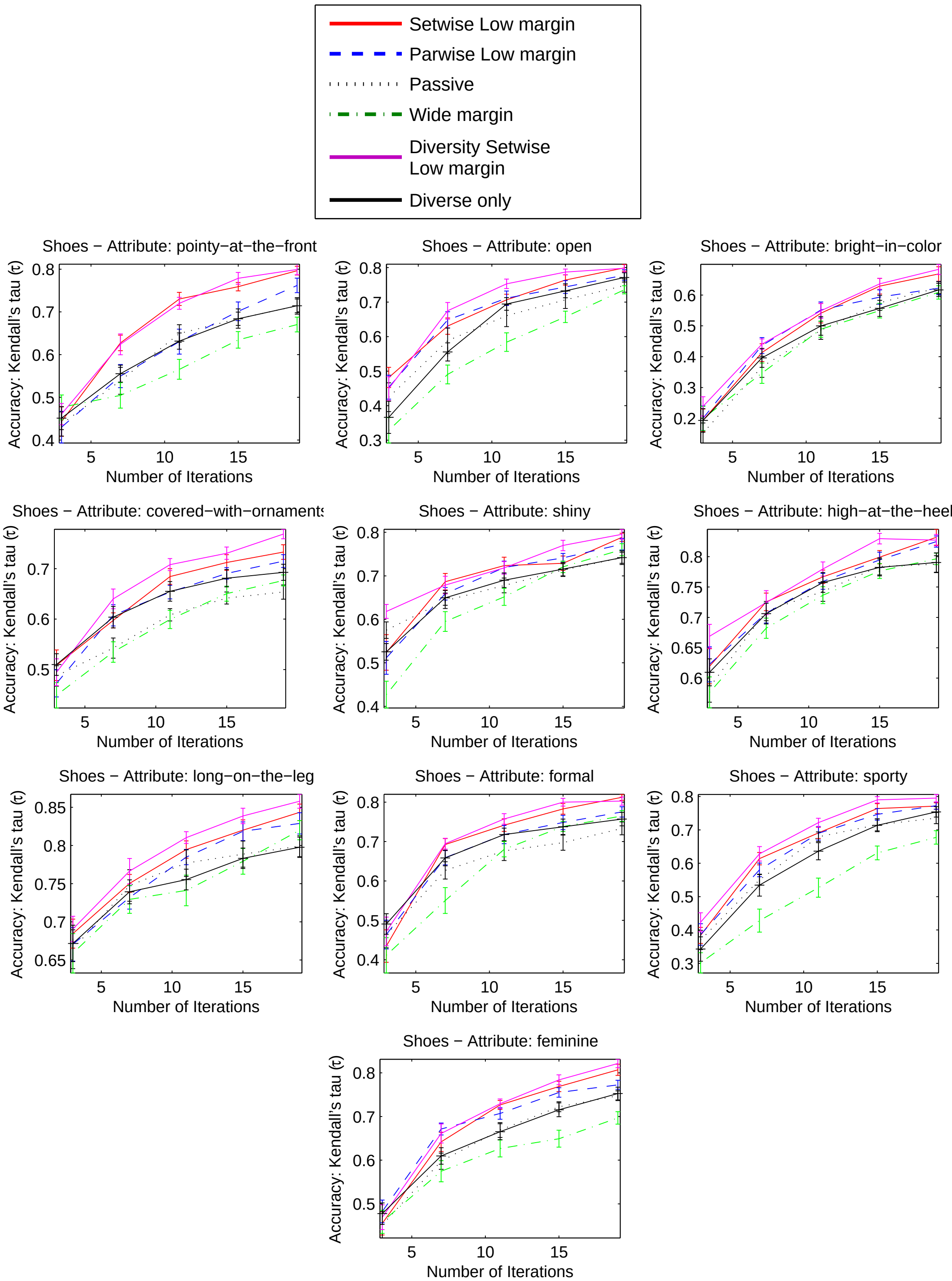

Figure 2: Per attribute learning curves for the Shoes dataset, offline experiments.

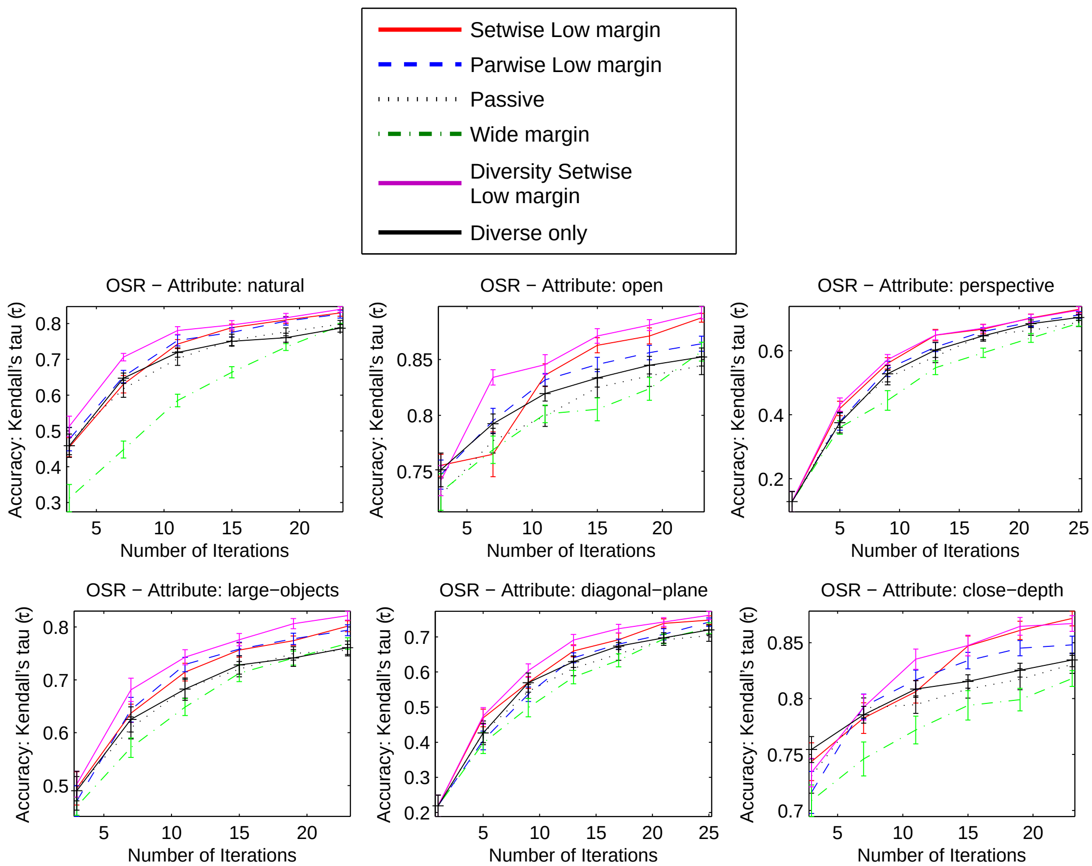

Figure 3: Per attribute learning curves for the OSR dataset, offline experiments.

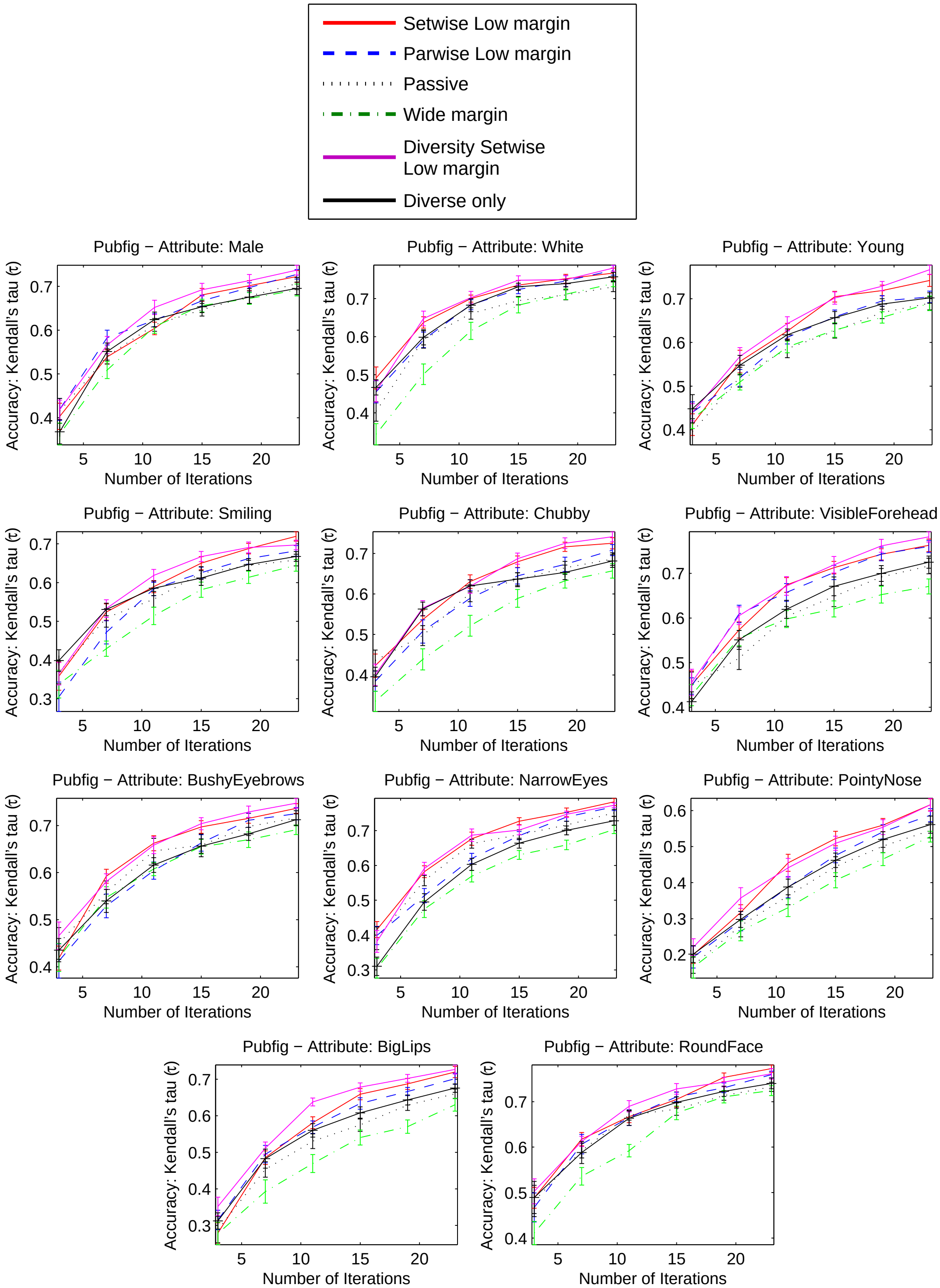

Figure 4: Per attribute learning curves for the PubFig dataset, offline experiments.

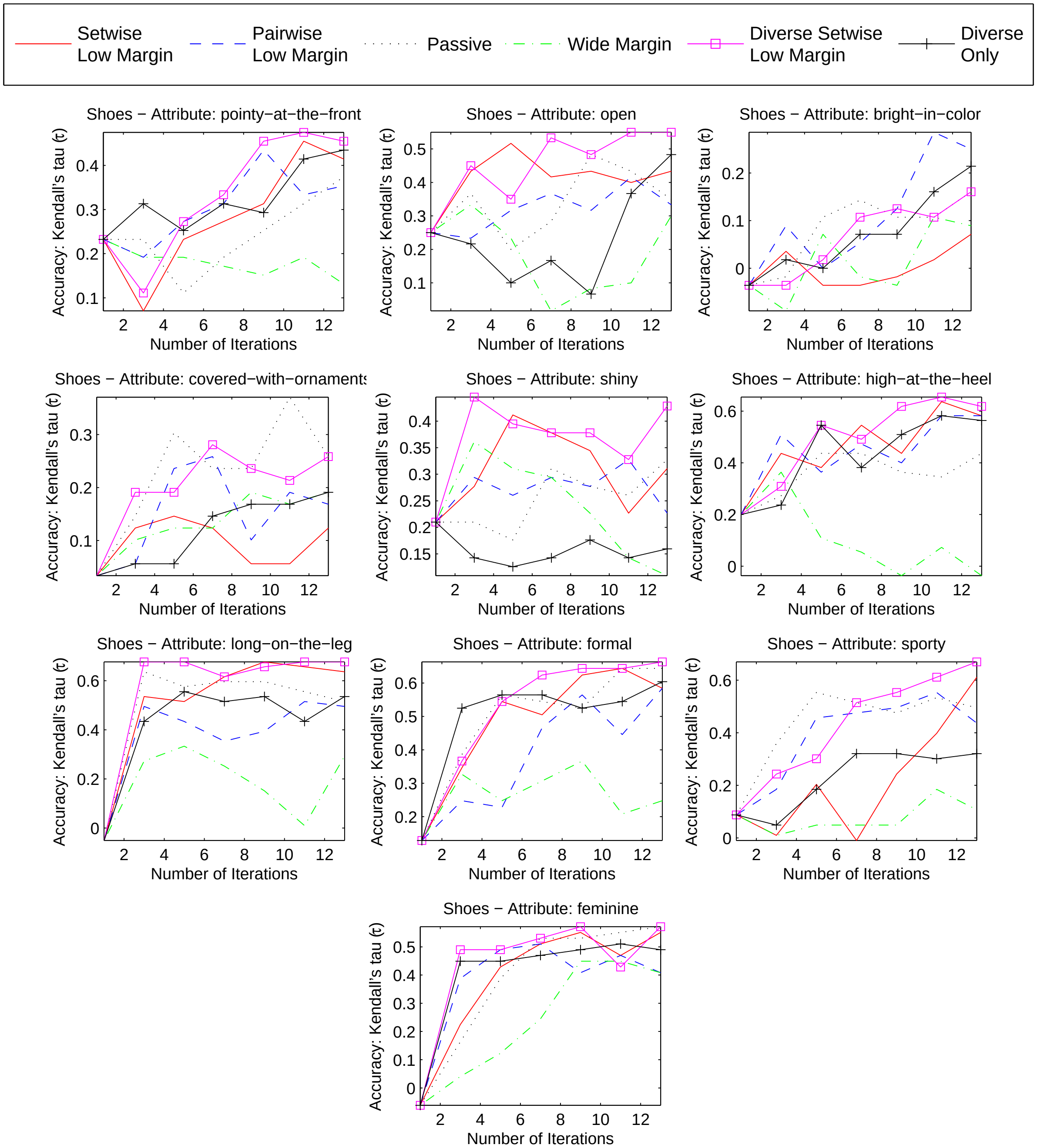

Figure 5: Per attribute learning curves for the Shoes dataset, live experiment.

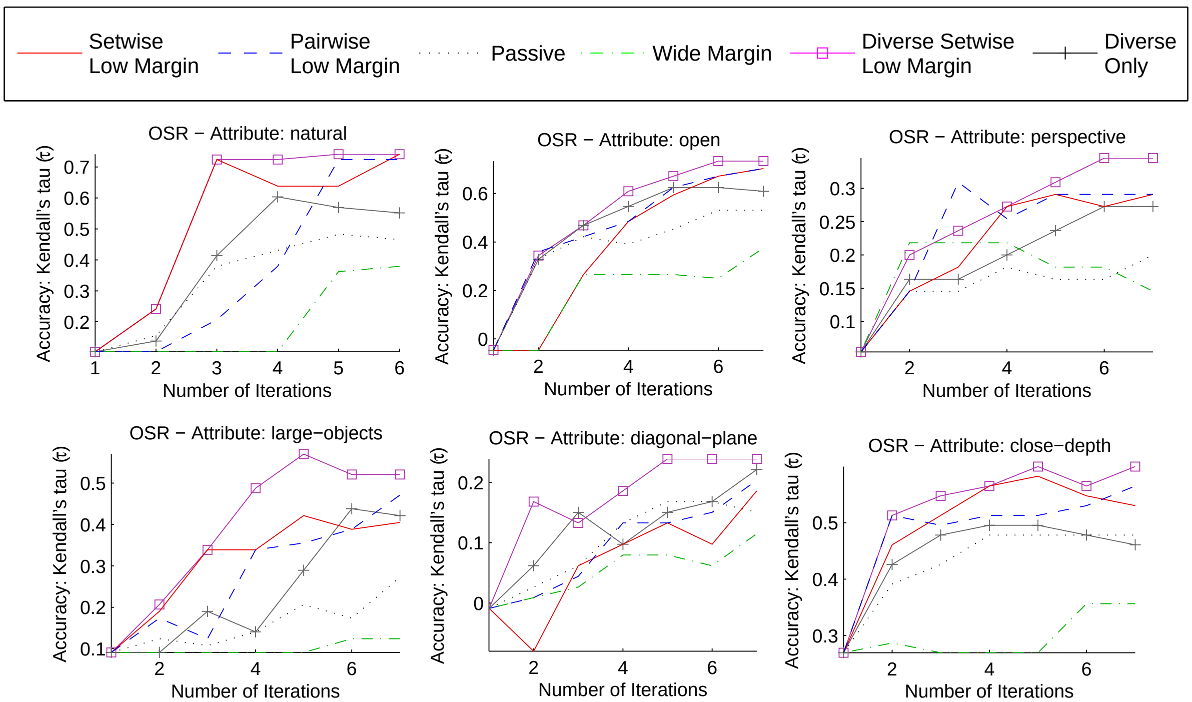

Figure 6: Per attribute learning curves for the OSR dataset, live experiment.

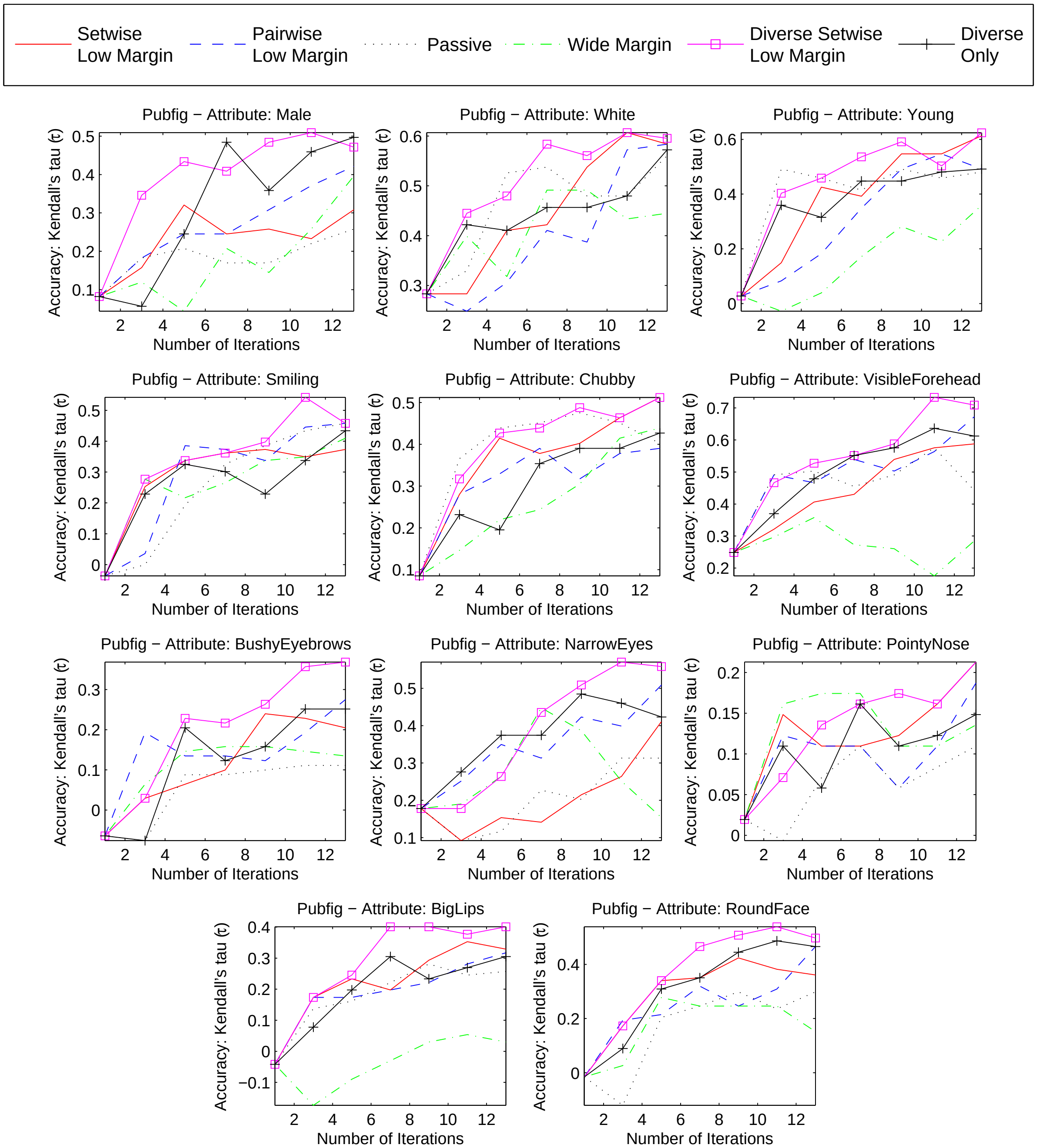

Figure 7: Per attribute learning curves for the PubFig dataset, live experiment.