Algorithm 1 Token PruMerge and PruMerge+ algorithms for reducing the number of visual tokens.

```
Require: Key and Query matrices of ViT's penultimate layer, $\mathbf{K} = \{\mathbf{k}_1, \cdots \mathbf{k}_n\}$ and $\mathbf{Q} = \{\mathbf{q}_1, \cdots \mathbf{q}_n\}$. The penultimate layer's output tokens, $\mathbf{Y} = \{\mathbf{y}_1, \cdots \mathbf{y}_n\}$. $n$ is the number of input visual tokens.
Ensure: Refine $\mathbf{Y}$ to $m$ (adaptive) visual tokens $\mathbf{Y}' = \{\mathbf{y}'_1, \cdots \mathbf{y}'_m\}$, in which $m \ll n$.
Token PruMerge:
Calculate attention between visual token and class token $\mathbf{a}_{[cls]}$ using Equation 2.
Use the outlier detection algorithm IQR to adaptively select $m$ important visual tokens' indices $\{i_1, \cdots, i_m\}$ based on $\mathbf{a}_{[cls]}$ (see Sec. 3.2).
(Optional for PruMerge+, see Sec. 3.4) Calculate the outlier ratio $r_o = \frac{m}{n}$.
(Optional for PruMerge+) Spatial-uniformly sample visual tokens with $r_o$, and get $\{i_{1+m}, \cdots, i_{2m}\}$.
(Optional for PruMerge+) Update the selected tokens' index with $\{i_1, \cdots, i_m, i_{m+1}, \cdots, i_{2m}\}$.
for $p = \{i_1, \cdots, i_m\}$ do (see Sec. 3.3)
    Calculate the distance between selected token $\mathbf{y}_p$ and other visual tokens, $\mathbf{y}_{\{1,\cdots,n\}/p}$;
    Use $\mathbf{y}_p$ as cluster center and find the $k$ most similar tokens, with indices $\{j_1, \cdots, j_k\}_p$;
    Update cluster center token with weighted sum: $\mathbf{y}'_p = \sum_{q=1}^{k} \mathbf{a}[j_q] \cdot \mathbf{y}_{j_q}$;
end for
Output a refined stack of visual tokens $\mathbf{Y}' = \{\mathbf{y}'_1, \cdots \mathbf{y}'_m\}$.
```

7. Appendix

7.1. PruMerge Algorithm

Algorithm 1 outlines the complete procedures for the PruMerge and PruMerge+ algorithms, both implemented at the end of the visual encoder. Our algorithms are implemented at the end of visual encoder. The first step leverages the sparsity in class attention to guide the selection of the most informative tokens. In the second step, these selected tokens act as cluster centers to group similar tokens, which are then merged into their respective centers. This process constitutes the PruMerge algorithm. To increase the number of tokens in the refined output, the PruMerge+ algorithm further concatenates additional spatial tokens.

7.2. PruMerge Selected Token Visualization

We visualized the positions of the tokens selected by PruMerge in Fig.5. As shown, our token selection method ensures a comprehensive and representative sampling of visual tokens, which minimizes performance loss while achieving substantial token reduction.

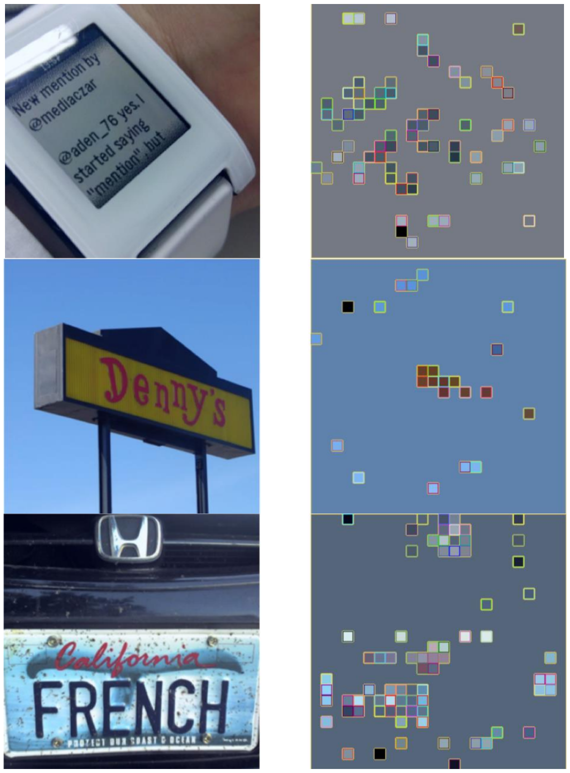
New mention by @mediaczar
@aden_76 yes,I started saying "mention" but
Denny's
California
FRENCH

Figure 5. Positions of the Selected Tokens