

INTRODUCTION

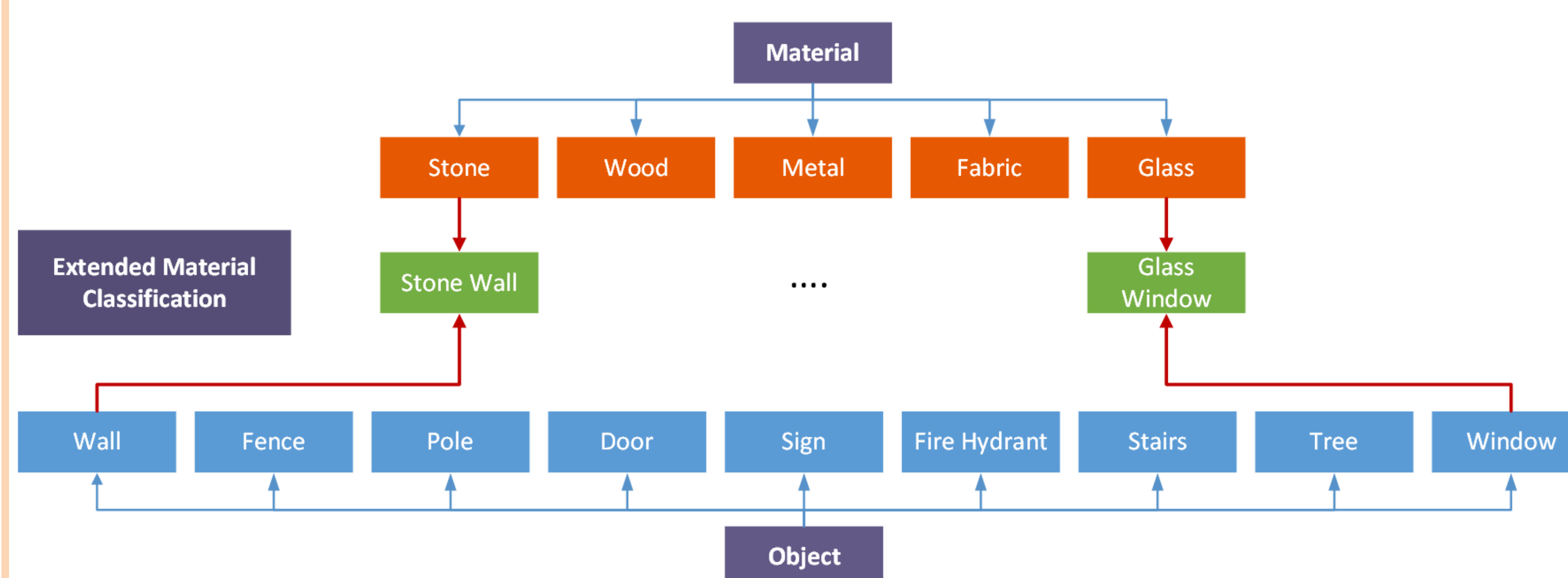
- **Recognizing surface material** of an object is important for developing various **smart city** solutions.
- Sometimes, materials of objects in an image are obstructed by incidental “covers” (referred to as **covered materials**). Examples of such covers include graffiti drawn by people or property damages caused by natural disasters.



- **Challenge:** Visual characteristics are distorted so surface material identification and classification is hard.

MATERIAL CLASSIFICATION

As a case study, we studied recognizing covered materials by graffiti using 19K labeled images provided by City of Los Angeles.

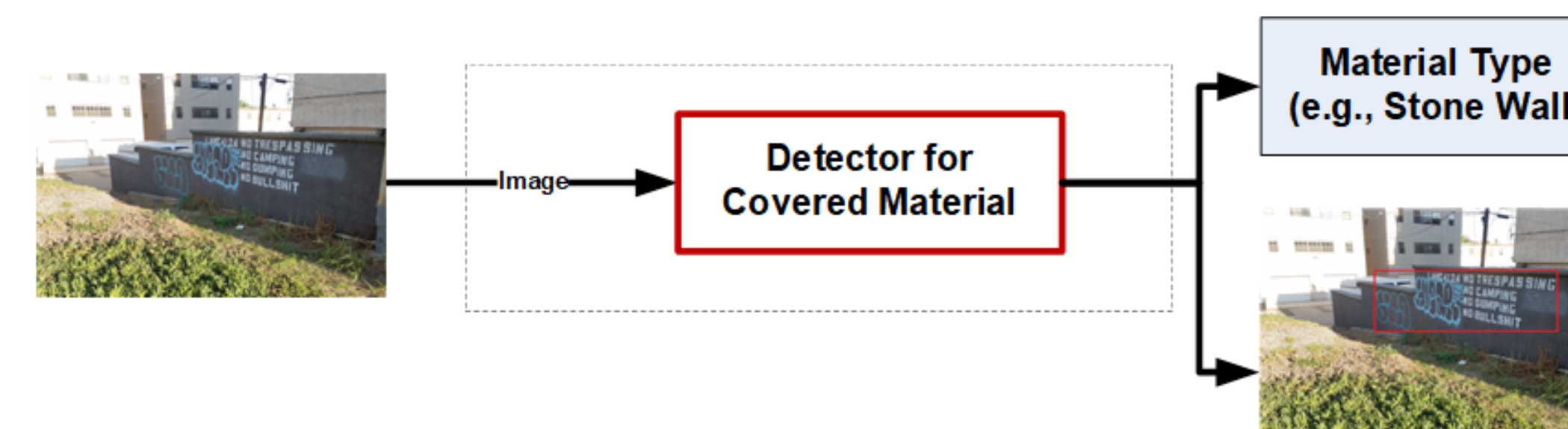


- Given t types of materials (i.e., $\mathbf{M} = \{m_1, m_2, \dots, m_t\}$) and n types of objects (i.e., $\mathbf{O} = \{o_1, o_2, \dots, o_n\}$), the extended material classification includes at maximum $t \cdot n$ materials.

LEARNING APPROACHES

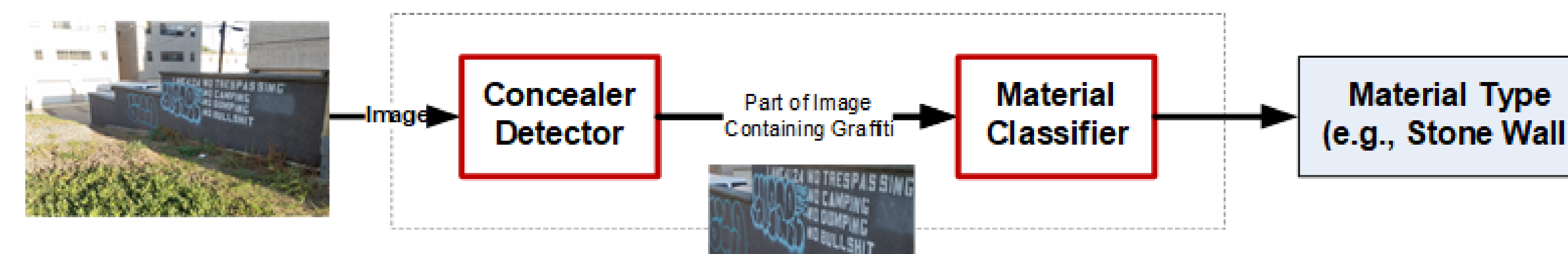
One-Phase Learning Approach (OLA)

- This approach is to consider a cover (e.g., graffiti) with its corresponding material as one unified object.



Two-Phase Learning Approach (TLA)

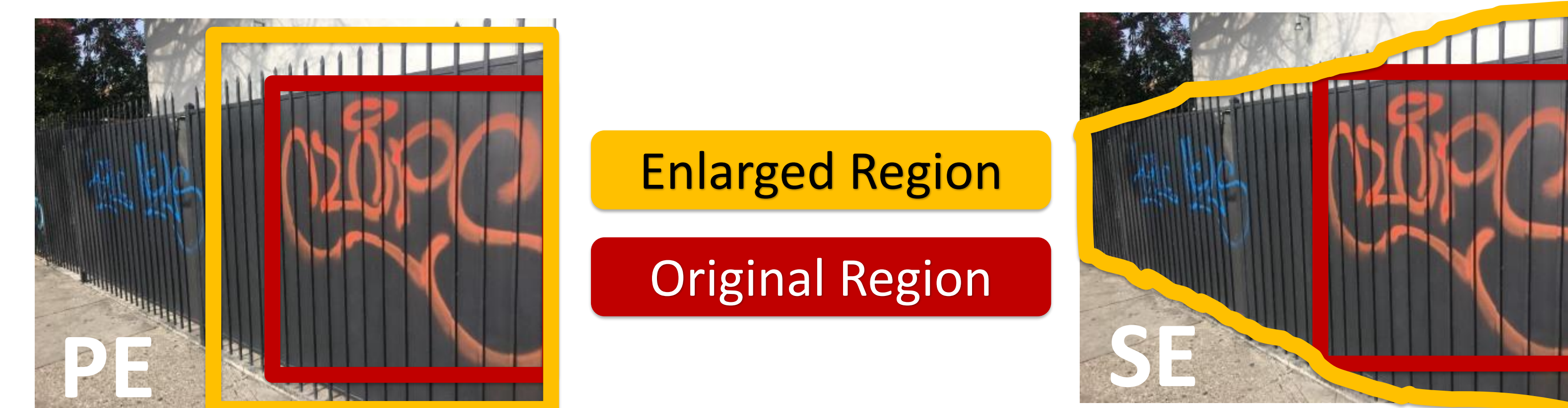
- First detect a cover and then classify the material covered by the detected cover.



APPROACHES W/ HEURISTIC EXPANSION

Proportional Expansion (PE)

- enlarge the learning region (which contains a covered material) by a factor (e.g., 1.3x).



Semantic Expansion (SE)

- Use a segmentation algorithm to enlarge a learning region to the segment containing the covered material.

BASELINE APPROACH

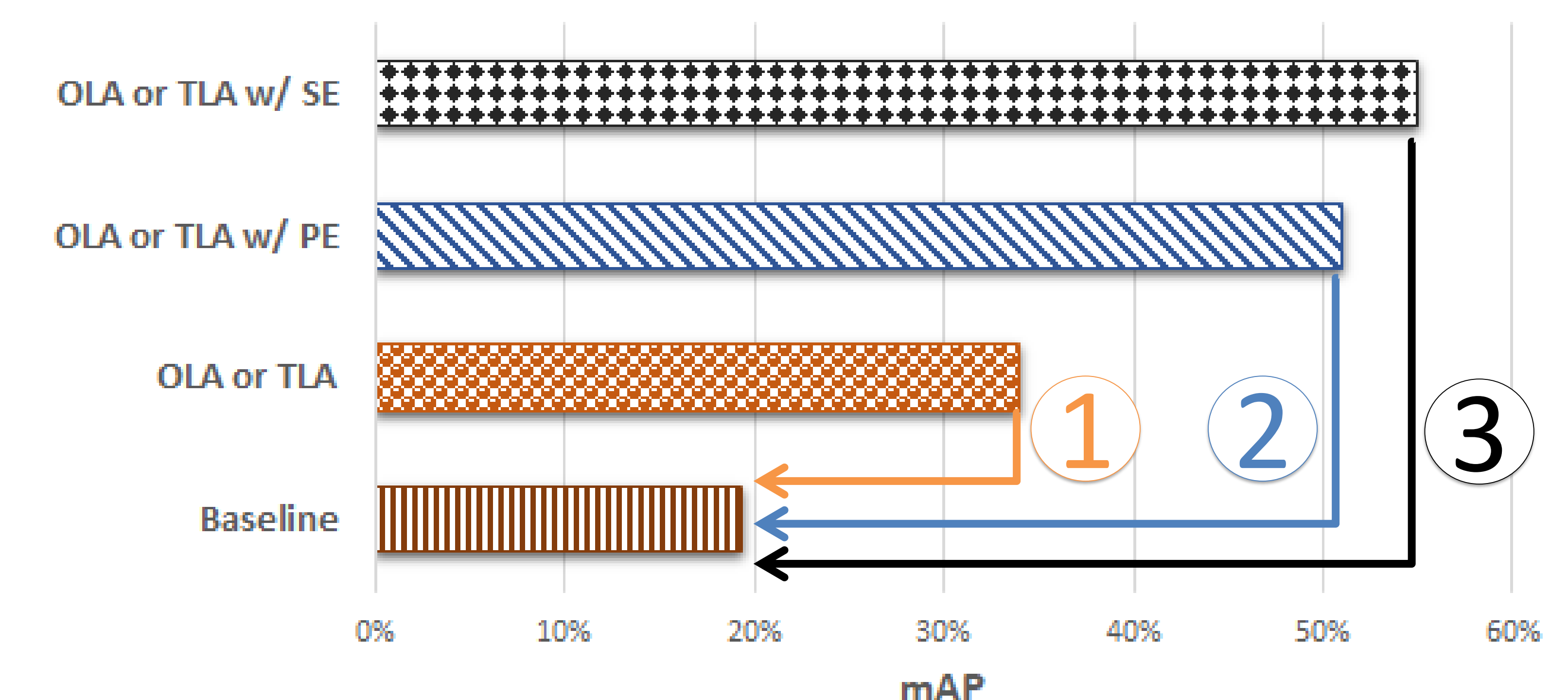
- A material recognition model (adapted version from Bell et al. [CVPR'15]). The model is trained on image segments that display plain materials.



Image Segments of “Plain material”
= Blue Box – Black Box

BASELINE VS. PROPOSED APPROACHES

- Our approaches are superior to the baseline in recognizing surface materials in images with graffiti.



- OLA and TLA improved mAP by a factor of **0.75x**
- PE enabled improving mAP by a factor of **1.6x**
- SE enabled improving mAP by a factor of **1.8x**