A CONTINUAL LEARNING APPROACH FOR LOCAL LEVEL ENVIRONMENTAL MONITORING IN LOW-RESOURCE SETTINGS

- Arijit Patra
- Siva Chamarti

- University of Oxford

Motivation: Crowdsourcing environmental monitoring

- Local monitoring – first line of defence against environmental manipulation
- Direct human monitoring is challenging due to terrain, logistics and availability of manpower
- Automated monitoring using sensors, and cameras may offer an alternative

Extended time monitoring

- Environmental events are temporally spaced and dynamically evolve
- Standard computer vision/deep network pipelines suffer from 'catastrophic forgetting' and show poor performance statistics on sequential adaptation under prior data unavailability
- Requirement of robust detection performance on deployment

- Solution: Continual learning strategies for sequential environmental monitoring tasks

Task schedule

- Task 1: Deforestation imagery detection
- Data curated from open source stock images;
- 4050 frames ranging from those sourced from tropical vegetation, deciduous forests, alpine forests, temperate shrublands and equatorial foliage
- Validation on holdout set of forestry scenes of ecological regions in Low and Middle Income Countries (LMIC).

- Task 2: Forest fire detection
- A set of 2000 images for the incremental task
- No. of frames: 600 with smoke, 500 with observable flames, 900 without smoke or fire
- Validation on both new task holdout set and on old task holdout set

Methodology

- A SqueezeNet, MobileNet and a MobileNet v2 backbone is used with the convolutional stack separated to process the image frames and associated modalities (such as log mel spectrograms for audio input if available).

- After final convolutional stages, feature maps are flattened and concatenated to obtain a joint representation vector which feeds to a cross-entropy objective at initial training:

$$L_C(y,p) = -\sum_{i=1}^{K_1} y_i \cdot \log(p_i)$$

- The pre-softmax neurons are retained and averaged per-class so as to serve as class-specific 'logits' that are weighted and summed up obtain the old classes' representation

$$z_{old} = \sum_{i=1}^{K_1} w_i z_i$$

- Summation weights ($w_1, w_2, \ldots, w_{k1}$) are calculated as inverse of class-specific AUC on the validation data for the initial Stage 1 classes.

- This averaged representation serves as a regularizer in a knowledge distillation loss during the incremental training, which uses a cross-entropy with labels for the new classes, and the distillation term for providing the model a 'snapshot' of the past tasks

$$L_D(z_{old}, \hat{y}) = -\sum_{i=1}^{N} softmax(\frac{z_{old}}{T}) \cdot \log(softmax(\frac{\hat{y}_i}{T}))$$

- Then, the overall objective during incremental training becomes…

$$L = \lambda L_C + (1-\lambda) L_D$$

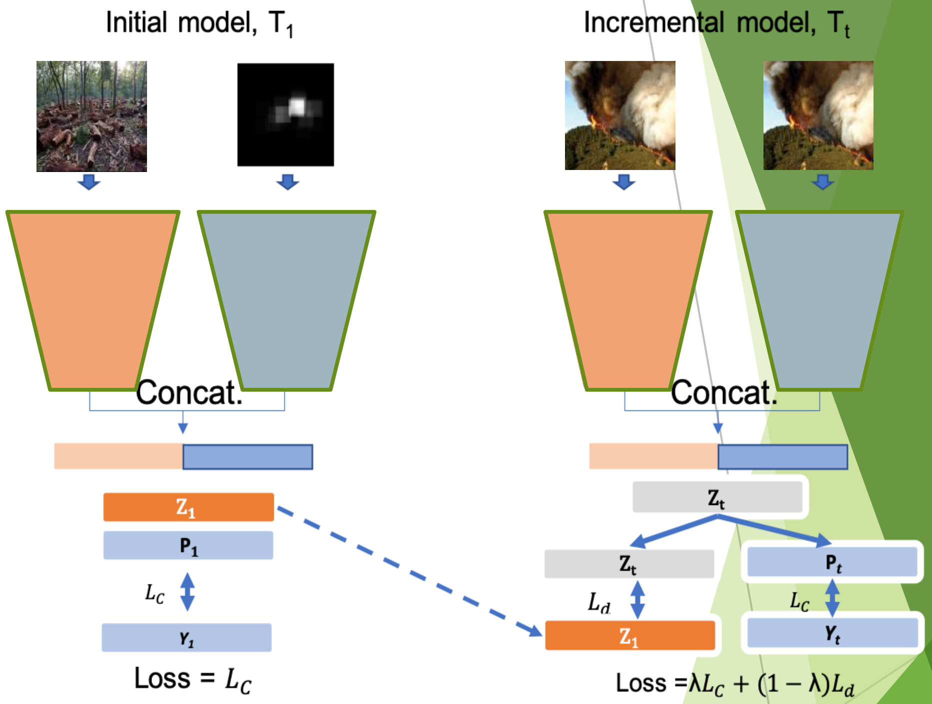

Fig. 1. The initially trained model (on Stage 1 tasks) is later trained for an incremental task at Stage t (here, t = 2), with cross-distillation using logits stored from initial stages.

Results

- For training, we start with the initial task (Task 1: forestry) with the cross entropy objective, and progress to the incremental task (Task 2: forest fire detection) with a joint distillation and cross-entropy regime
- Data augmentation was applied with vertical and horizontal flips,and random cropping
- The training for initial stages is performed over batches of 100 frames in 500 epochs, with a learning rate of 0.001 and a logistic regression objective for bounding box regression along with a cross-entropy loss term for the classification part
- The MobileNetv2 implementation was 6x faster than the SqueezeNet backbone detector and 3.5x faster than the one using MobileNet, demonstrating the efficiency gains through group convolution based models

Table 1: Evolution of model performance over the first task of deforestation monitoring. The average percentage classification accuracy for individual classes are presented

	Machinery	*Reduced cover*	*Untouched cover*
SqueezeNet + YOLO)	0.68	0.64	0.61
MobileNet + YOLO	0.81	0.77	0.70
MobileNetv2 + YOLO	0.83	0.79	0.75

Table 2: Model performance in terms of overall accuracy %age, over successive stages of tasks addition, with and without distillation based incremental learning for MobileNet v2 + YOLO

	Stage 1: Deforestation	*Stage 2: Wildfire monitoring*
MobileNetv2 + YOLO with IT	0.80	0.73
MobileNetv2 + YOLO	0.79	0.67

Thank you