LID Challenge: Weakly Supervised Semantic Segmentation

3d place solution

NoPeopleAllowed: The 3 step approach to weakly supervised semantic segmentation

Mariia Dobko, Ostap Viniavskyi, Oles Dobosevych
UCU & SoftServe team
The Machine Learning Lab at Ukrainian Catholic University, SoftServe

Outline

- Problem description
- Competition
- Approach architecture
  - Step 1. CAM generation via classification
  - Step 2. IRNet for CAM improvements
  - Step 3. Segmentation
- Postprocessing
- Results
- Conclusions

Problem description

A key bottleneck in building a DCNN-based segmentation models is that they typically require **pixel level annotated images** during training. Acquiring such data demands an **expensive**, and **time-consuming** effort.

Image-level annotations

15 times faster to label

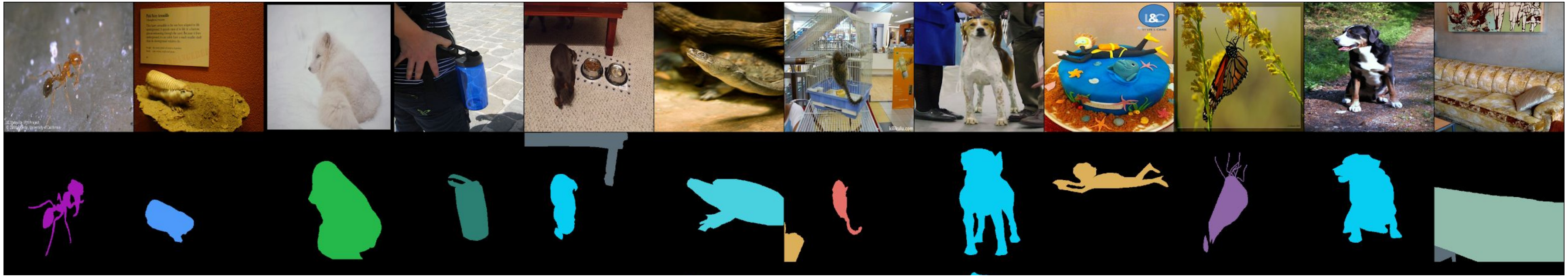
> 25 times cheaper
0.035$ per image for class,
3.45$ for segmentation

We develop a method that has a **high performance** in segmentation task while also **saves time and expenses** by using only **image-level annotations**.

LID Challenge Dataset

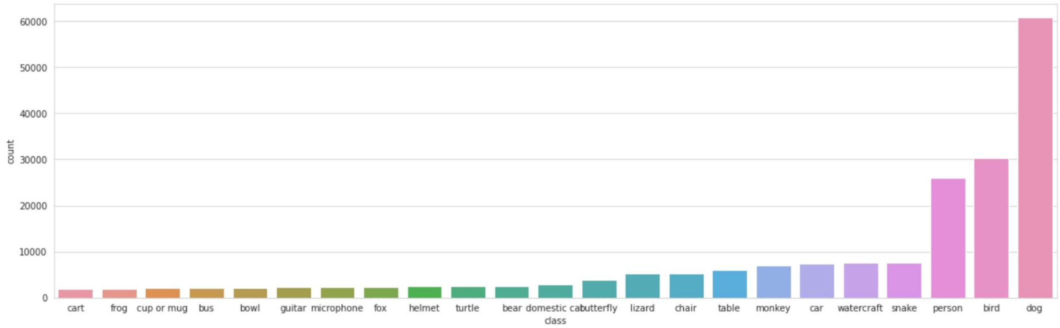
- **Multilabel multiclass**
- Pixel-wise labels are provided for validation set only
- **No pixel-wise annotations** are allowed for training

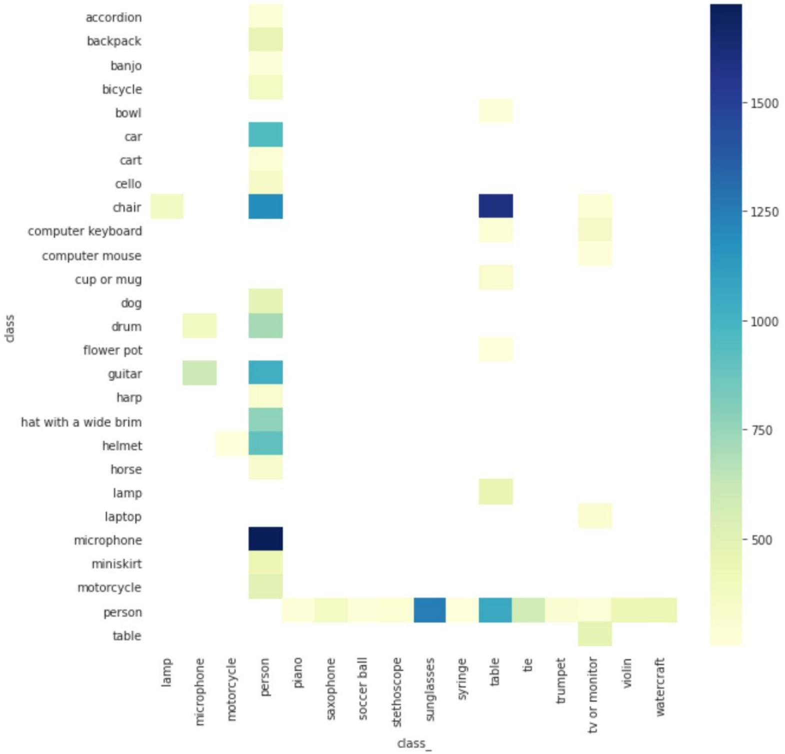
- **200** classes + background
- **456,567** training images
  - validation: **4,690**
  - test: **10,000**



Challenges

- **High imbalance** in classes: ‘person’, ‘bird’, ‘dog’
- **Missing labels**
- Subset of 2014 has better labels for ‘person’, than the whole dataset





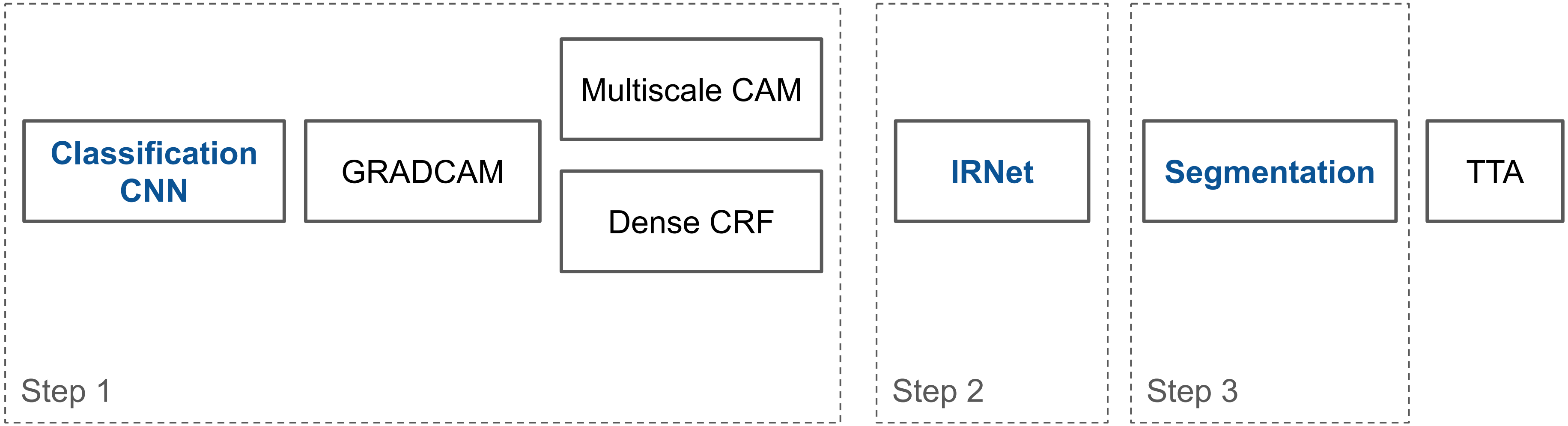
Previous works

Expectation-Maximization methods
Multiple Instance Learning methods
Object Proposal Class Inference methods
Self-Supervised Learning methods

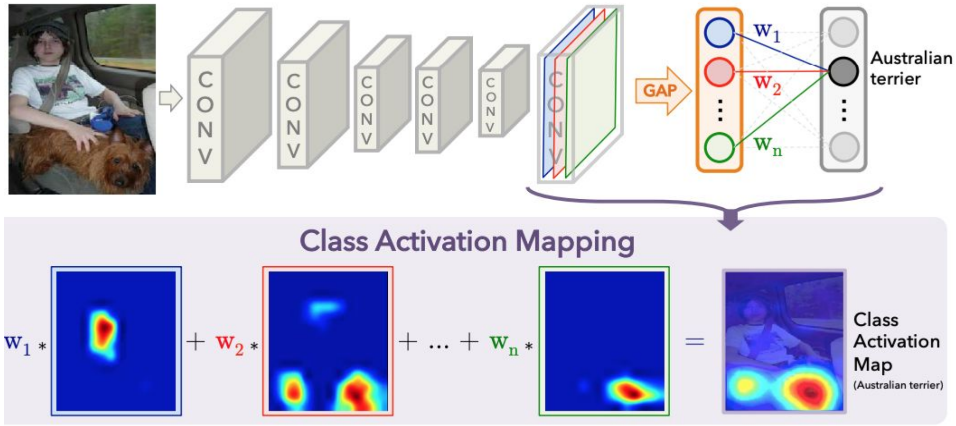
Method	Year	Code available?	Train/test code	Code framework	VOC2012 mIoU (%) *val*	VOC2012 mIoU (%) *test*
MIL-FCN (Pathak et al., 2014)	2015	Y	Train/test	MatConvNet	25.7	24.9
CCNN (Pathak et al., 2015)	2015	Y	Train/test	Caffe	35.3	35.6
EM-Adapt (Papandreou et al., 2015)	2015	Y: Caffe, TensorFlow	Train/test	Caffe, TensorFlow	**38.2**	**39.6**
DCSM w/o CRF (Shimoda and Yanai, 2016)	2016	Y	Test	Caffe	40.5	41
DCSM w/ CRF (Shimoda and Yanai, 2016)	2016	Y	Test	Caffe	44.1	45.1
BFBP (Saleh et al., 2016)	2016	N	No	-	46.6	48.0
SEC (Kolesnikov and Lampert, 2016b)	2016	Y: Caffe, TensorFlow	Train/test	Caffe, TensorFlow	**50.7**	**51.7**
WILDCAT + CRF (Durand et al., 2017)	2017	Y	Train/test	PyTorch	43.7	-
SPN (Kwak et al., 2017)	2017	Y	Custom layer only	Keras	50.2	46.9
AE-PSL (Wei et al., 2017)	2017	N	No	-	**55.0**	**55.7**
PRM (Zhou et al., 2018)	2018	Y	Test	PyTorch	53.4	-
DSRG (VGG16) (Huang et al., 2018)	2018	Y: Caffe, TensorFlow	Train/test	Caffe, TensorFlow	59.0	60.4
PSA (DeepLab) (Ahn and Kwak, 2018)	2018	Y	Train/test	PyTorch	58.4	60.5
MDC (Wei et al., 2018)	2018	N	No	-	60.4	60.8
DSRG (ResNet101) (Huang et al., 2018)	2018	Y: Caffe, TensorFlow	Train/test	Caffe, TensorFlow	61.4	63.2
PSA (ResNet38) (Ahn and Kwak, 2018)	2018	Y	Train/test	PyTorch	**61.7**	**63.7**
FickleNet (Lee et al., 2019)	2019	N	No	-	61.2	61.9
IRNet (Ahn et al., 2019)	2019	Y	Train/test	PyTorch	**63.5**	**64.8**

Chan et al. *A Comprehensive Analysis of Weakly-Supervised Semantic Segmentation in Different Image Domains*

Our approach architecture



Step 1. CAM generation via classification



Zhou et al. *Learning deep features for discriminative localization*

Input

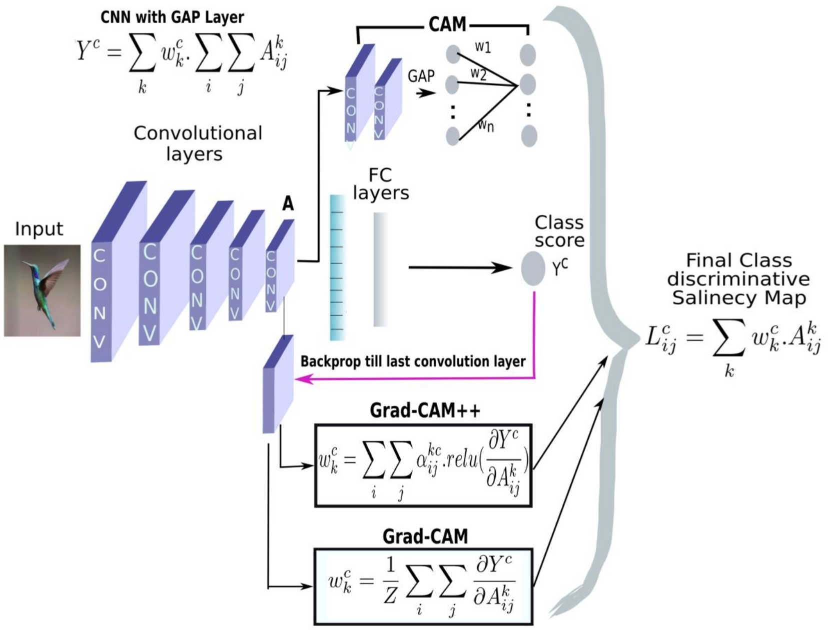
- 72k - train, 12k validation
- balanced dataset
- no person class

Results

Step 1. CAM generation via classification

Tested approaches

- ResNet50 vs. VGG16 → ResNet produces artifacts
- VGG16 with additional 4 conv layers
- GRADCAM vs. GRADCAM++ → GRADCAM++ usually gives just slightly better results



Chattopadhyay et al. Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks

Step 2. IRNet for CAM improvements

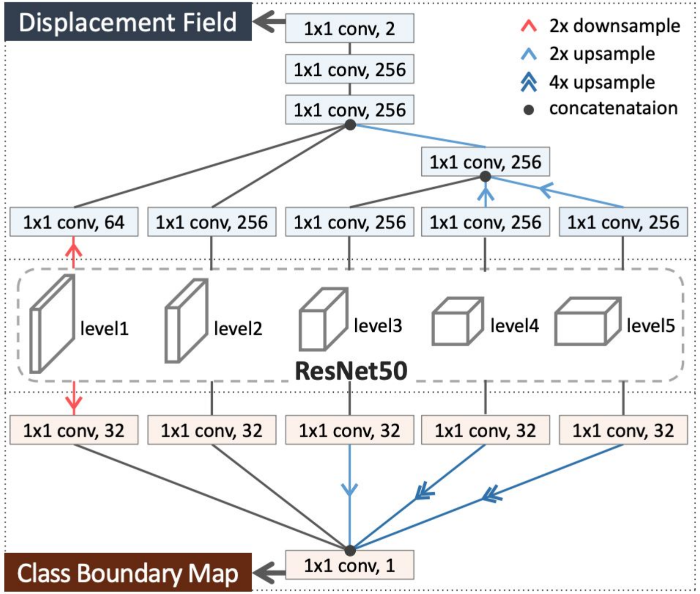


Figure 2. Overall architecture of IRNet.

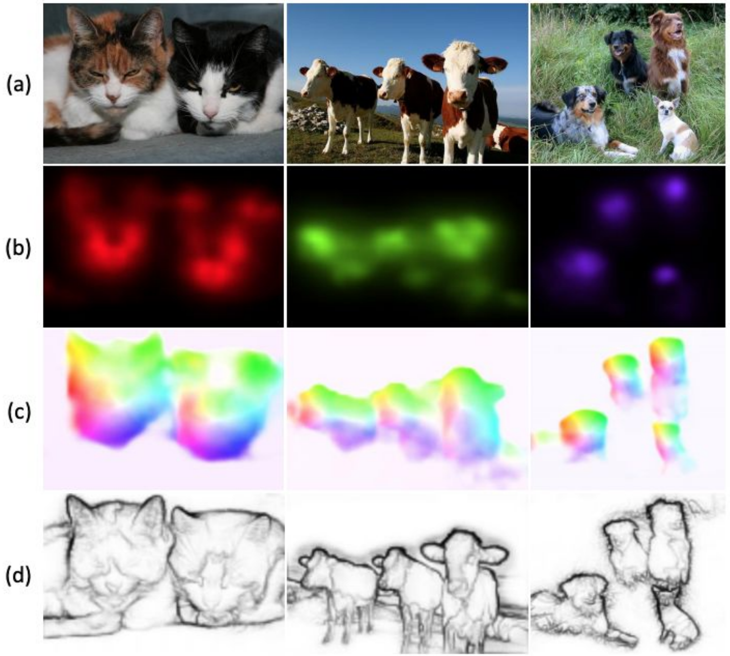
Ahn et al. *Weakly supervised learning of instance segmentation with inter-pixel relations.*

Input

- Select most confident maps
- Threshold CAMs into confident BG, confident FG and unconfident regions

Results

IRNet



Ahn et al. *Weakly supervised learning of instance segmentation with inter-pixel relations.*

IRNet's two branches:
1 - learns the displacement field
2 - learns class boundaries

$$\mathcal{L} = \mathcal{L}_{\text{fg}}^{\mathcal{D}} + \mathcal{L}_{\text{bg}}^{\mathcal{D}} + \mathcal{L}^{\mathcal{B}}.$$

Losses for Displacement fields (foreground & background)

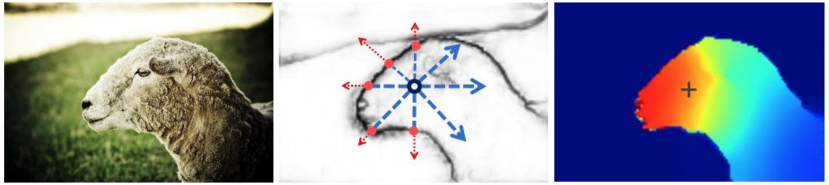
Loss for class boundary detection

IRNet. Class Boundary Detection

$$a_{ij} = 1 - \max_{k \in \Pi_{ij}} \mathcal{B}(\mathbf{x}_k)$$

(d)

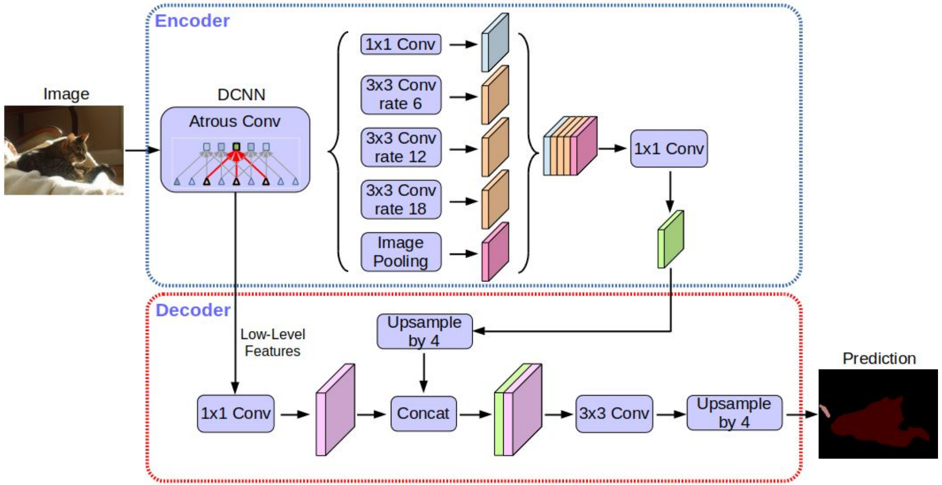
$$\mathcal{L}^{\mathcal{B}} = - \sum_{(i,j) \in \mathcal{P}^+_{\text{fg}}} \frac{\log a_{ij}}{2|\mathcal{P}^+_{\text{fg}}|} - \sum_{(i,j) \in \mathcal{P}^+_{\text{bg}}} \frac{\log a_{ij}}{2|\mathcal{P}^+_{\text{bg}}|} - \sum_{(i,j) \in \mathcal{P}^-} \frac{\log(1 - a_{ij})}{|\mathcal{P}^-|}$$



Ahn et al. *Weakly supervised learning of instance segmentation with inter-pixel relations.*

Step 3 - Segmentation

DeepLab v3+



Chen et al. *Encoder-decoder with atrous separable convolution for semantic image segmentation.*

Input

- 352x352 input images
- Strong augmentations
- ~42k images for training

Results

Postprocessing

Test Time Augmentations are added after segmentation step. The combination of 2 types of different TTAs, with one having 3 parameters, result in total 6 predictions, which are averaged by mean.

Secret insights

- **VGG** is better for CAM generation as **ResNet** gives artifacts
- **Decrease the output stride** of VGG by removing some of the max pooling operations
- **Confident** and **unconfident** regions for IRNet
- **Multiscale CAM** give a large improvement
- **Dense CRF** doesn’t require training, helps to rectify boundaries
- **TTA** after segmentation step drastically improves the results
- Replace stride with dilation in DeepLabv3+ to **decrease the output stride**

Metrics

Classification Quality

- **F-1 score**

$$F1 = \frac{2 \times precision \times recall}{pression + recall}$$

Step 1. Classification

Segmentation Quality

- **Mean IoU**

$$mIoU = \frac{1}{k}\sum_{i=1}^{k}\frac{TP_{ii}}{\sum_{j=1}^{k} FN_{ij} + \sum_{j=1}^{k} FP_{ij} - TP_{ii}}$$

- **Pixel Accuracy**
- **Mean Accuracy**

Step 2-3. IRnet & Segmentation

Quantitative Results

Model	IRNet threshold	TTA	Person CAM	Mean IoU
DeepLabv3+ encoder: **ResNet50**	**0.3**	No	No	36.65
		Yes	No	**39.64**
		Yes	**Yes**	**39.80***
	0.5	No	No	37.11
		Yes	No	39.58
DeepLabv3+ encoder: ResNet101	0.5	No	No	36.14
		Yes	No	37.15

* wasn't submitted

Validation set

Experiments with different architectures and parameters on the 3rd step

Quantitative Results

Test set:

DeepLabv3+
+
TTA
(Horizontal Flip,
Multi-scaling)

Rank	Participant team	Mean IoU	Mean accuracy	Pixel accuracy	Last submission at
1	cvl	45.18	59.62	80.46	1 day ago
2	VL-task1	37.73	60.15	82.98	2 days ago
3	UCU & SoftServe	37.34	54.87	83.64	2 days ago
4	IOnlyHaveSevenDays	36.24	68.27	84.10	2 days ago
5	play-njupt	31.90	46.07	82.63	1 month ago
6	xingxiao	29.48	48.66	80.82	1 month ago
7	hagenbreaker	22.50	39.92	77.38	19 days ago
8	go-g0	19.80	38.30	76.21	20 days ago
9	lasthours-try	12.56	24.65	64.35	1 day ago
10	WH-ljs	7.79	16.59	62.52	2 days ago

Open questions

Different types of **regularization** added to the first step ⟶ Improve the **classification**

Downsampling was used to balance data ⟶ **Upsampling** or **combination** of both should be tested

Adding person class labels to the other steps of pipeline ⟶

Ability to provide better results for a class which is highly present in data, though severely mislabeled

Mean IoU per class allows to obtain high score even when some classes are skipped ⟶

A different metric or combination of metrics should be chosen as a premier for this task

Thank you for attention!

presentation

dobko_m@ucu.edu.ua

viniavskyi@ucu.edu.ua

dobosevych@ucu.edu.ua