Guide Me: Interacting with Deep Networks
[CVPR 2018]

Christian Rupprecht, Iro Laina, Nassir Navab, Gregory D. Hager, Federico Tombari

Presenters:

Siba Smarak Panigrahi and Sohan Patnaik

Reading Session VII
Kharagpur Data Analytics Group
IIT Kharagpur

June 06, 2021

Important Notes

- We expect you all to have certain queries regarding the presentation, certain intricate doubts maybe... **Please put them in the chat of this meeting (if you feel shy) else unmute and speak**.
- Finally, a **Google Form** [Link] will be released for feedback but most importantly, we ask you to put up name of any AI-related paper to be presented in the upcoming sessions!

Link to GitHub Repo : Click Here
Link to join Slack Workspace : Click Here
Link to KDAG YouTube Channel : Click Here
Link to doc on good AI Blogs/Resources/Topics : Click Here

Outline

- Introduction
- Architecture Overview
- Methods
  - Guiding Block
  - Guiding by Back-propagation
  - Learning to Guide with Text
    - Training with queries
    - Generating queries
- Results
  - Performance after a number of questions
  - Location of the guiding block
  - Complexity of Hints and Guiding multiple times
- Conclusion

Let’s Have a look at the picture:

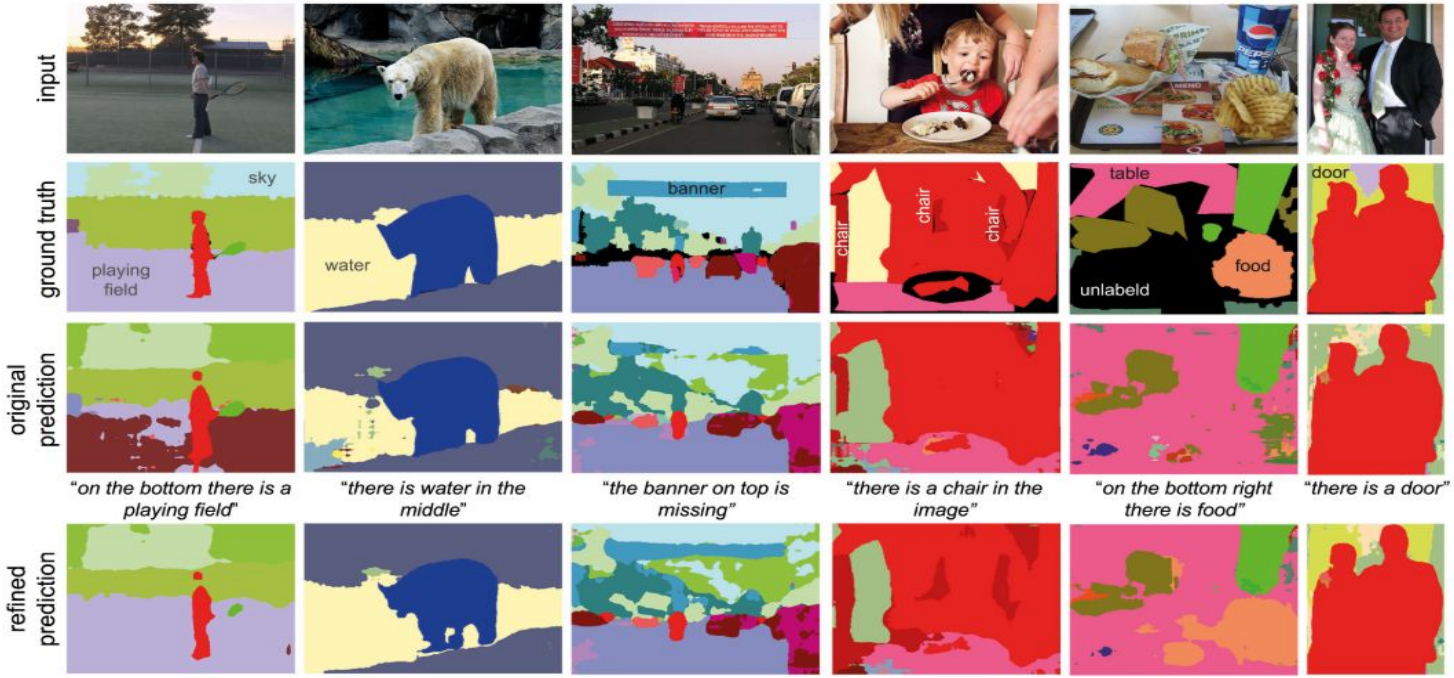

Figure 3. **Qualitative Results**. We show qualitative results using `find` hints for missing classes. In the first example, we resolve a confusion between `ground` and `playing field`. In the second example, we show that the often occurring spurious predictions can also be handled. The third column shows that the network get the hint to find the `banner`, although it bleeds slightly into the `building` below. In the fourth and the last column, classes that are heavily occluded can be discovered too after guiding. The black ground truth label stands for `unlabeled` thus any prediction is allowed there. Please see the supplementary material for additional examples.

Introduction

- This paper outlines the idea of interaction of humans with deep learning models to improve performance.
- A novel method was proposed to guide a pre-trained convolutional network through user input to improve its performance during inference
- This is highly useful in medical image analysis, computer assisted analysis etc.
- The core idea was to develop a guiding mechanism that translates user feedback into changes in the internal activations of the network, thus acting as a means of rethinking the inference process.

Architecture Overview

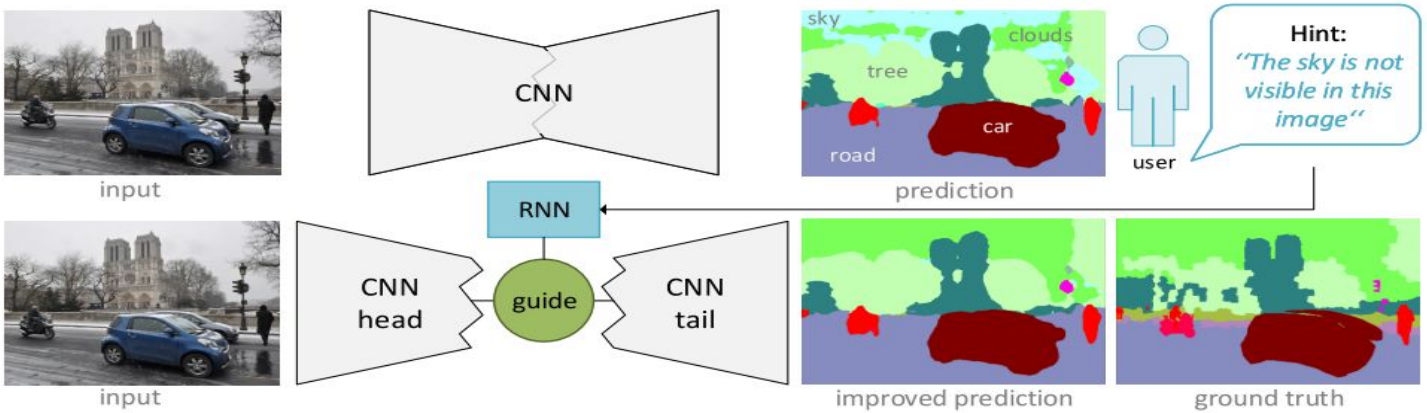

Figure 1. **Overview.** We introduce a system that is able to refine predictions of a CNN by injecting a guiding block into the network. The guiding can be performed using natural language through an RNN to process the text. In this example, the original network had difficulties to differentiate between the `sky` and the `cloud` classes. The user indicates that there is no sky and the prediction is updated, without any CNN weight updates and thus without additional training.

Guide : Interacts with the guided CNN through a guiding block adjusts activation maps of the CNN

Guided CNN is thus split into two parts:

- **head (h)** : processes the input until it reaches the guiding block
- **tail (t)** : rest of the guided network up to the final prediction layer.

Thus, output with given input x can be written as $\tilde{y} = t(h(x))$.

Hint : The information that the guide uses to modify the guided network

Methods

A. Guiding Block

- The interaction module is integrated into a fixed CNN with two different approaches:
  - guiding with user clicks and back-propagation
  - natural language inputs
- The network encodes specific features in each channel
- The guide is designed so as to strengthen or weaken the activations per channel
- This emphasizes or suppresses information, which results in a prediction in a semantically meaningful way.

$$A'_c = (1 + \gamma_c^{(s)})A_c + \gamma_c^{(b)}$$

The head predicts a feature representation $h(x) = A \in \mathbb{R}^{HxWxC}$ with width W, height H and number of channels C.

- In the previous equation, it was only possible to alter or provide information across channels.
- No spatial information was captured.
- In order to capture spatial information and introduce new predictions driven by spatially localized changes, two new parameter vectors ∝ and β were introduced which were H and W dimensional respectively.
- This ensures some hint like : "On the top left you missed ..." can also be taken into account.
- The output fed into the tail is now given by the equation (pixel wise update in a channel)

$$A'_{h,w,c} = (1 + \alpha_h + \beta_w + \gamma_c^{(s)}) A_{h,w,c} + \gamma_c^b$$

- The vector equation is given as

$$y^* = t\left[(1 \oplus \alpha \oplus \beta \oplus \gamma^{(s)}) \odot h(x) \oplus \gamma^{(b)}\right]$$

where the tiling of the vectors α, β, γ along their appropriate dimensions is denoted with ⊕ and the Hadamard product with ⊙

B. Guiding by Back-propagation

- Network revises its prediction without additional learning. Why?
  - For a given sample x, we formulate an energy minimization task to optimize α, β and γ
  - Hint will be given as a sparse input $\hat{y}$ associated to a mask $\hat{m}$
  - $\hat{y}$ and $\hat{m}$ have the same dimensionality as the prediction $\tilde{y}$ and the ground truth y.
  - $\hat{m}$ is a binary mask that indicates the locations where a hint is given.

- In semantic segmentation, one can think of the hint as a single (or more) pixel(s) for which the user provides the true class – "**this [pixel] is a dog**" as additional information

- Originally, a certain loss **L {t (h(x)), y}** has been minimized during training of the network for a given task. We now optimize towards the same objective, **e.g. per-pixel cross entropy for segmentation**, but use the mask $\hat{m}$ to only consider the influence of the hint and **minimize for the guiding parameters, as opposed to the network's parameters**

$$\alpha^*, \beta^*, \gamma^* = \underset{\alpha,\beta,\gamma}{\operatorname{argmin}} \left[\hat{m} \odot \mathcal{L}(y^*, \hat{y})\right]$$

The story continues...

- Intuitively, the tendency of gradient descent to fall into local optima is desirable here. We are looking for the smallest possible α, β and γ that brings the guided prediction closer to the hint while avoiding degenerate solutions such as predicting the whole image as the hinted class.

- A user interaction scheme similar to the 20-question game of is set up.
- After an inference step, the network is allowed to ask the user for the class of a single pixel and the guiding layer updates the feature representation.

- The queried pixel is the one with the smallest posterior probability difference between the two most confident classes. This pixel has the highest interclass uncertainty, meaning that it is the most likely to flip.

C. Learning to Guide with Text

I. Training with queries

- Input query was encoded into word embedding matrix (Glove Embeddings), pretrained on large corpus
- Embedded words are fed to GRU at each time step.
- The guiding parameters $\propto$, β and γ were predicted as a linear mapping from the final hidden state of the GRU
- The prediction of fixed CNN and ground truth were fed into a hint generator (e.g. “the sky is not visible in this image”). **Hint generator will be discussed next in “Generating Queries”.**
- The standard loss for the given task (cross entropy loss) was reweighted giving positive weights to the classes mentioned in the query. This ensures only the hint “class” is taken into account or expressed.

II. Generating Queries

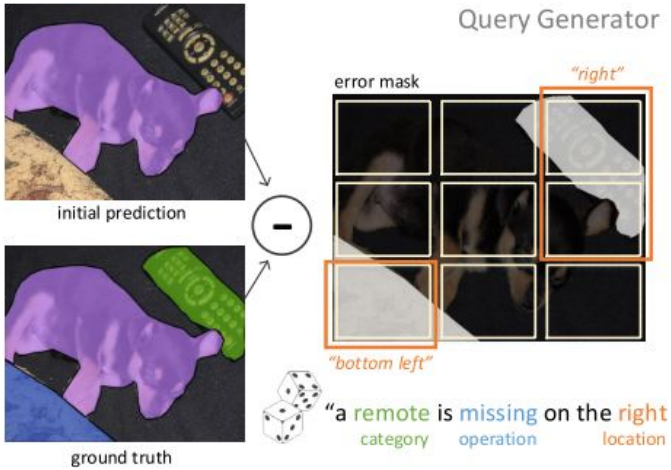

Figure 2. **Query Generator**. We illustrate the process to automatically generate queries to substitute the user during training.

- For generating queries, a combination of functionality, semantic categories and spatial layouts were used.
- The output prediction of the head was divided into NxN grid and for each grid cell, the erroneous classes were searched.
- Certain queries including 'find' and 'remove' were generated using a fixed template.
- E.g.: "There is a person in the top right" is a find query. Here, "there is a person" is the find query and "in the top right" specifies the spatial location in the image.
- It was observed that "**rmv**" yielded lower performance gain than "**find**". Why?

Results

Performance after a number of questions

#questions	0	1	5	10	15	20
FCN-8s						
mIoU	62.6	65.3	73.1	76.9	77.3	**81.0**
p.accuracy	91.1	91.8	94.1	95.3	96.0	**96.3**

This shows that with each user interaction the results improve (provision of hint).

Location of the guiding block

	res3a	res4a	res5a	res5c
mIoU	32.21	33.56	35.97	**36.50**

- The split position is chosen manually. But, a reasonable choice is the (spatially) smallest encoding that the network produces, as this layer likely contains the most high-level information in a compact representation.

- In general, a location that is very late - close to the prediction - inside the network often results in small, local changes in the output.

- Moving the block earlier results in more global changes affecting a bigger region and sometimes multiple classes. Early in the network, the feature maps, that it guides do not contain enough high-level information to guide appropriately.

Complexity of Hints & Guiding multiple times

hint complexity	guiding location res4a	res5a
`remove`	31.53	32.56
`find or rmv`	32.22	33.73
`find`	33.56	**35.97**

The table on the left shows results of using the two queries. It was observed that using only the “find” query at the guiding location res5a yielded better results.

The table on the right gives results of using multiple queries. It was observed that as the number of hints increased (more than 2), the network started to perform bad. Intuitively, we can say that we are telling everything to the network and it does not learn properly.

# hints	0	1	2	3	4
mIoU	30.53	34.04	**35.01**	34.24	31.44

Conclusion

- This paper proposed a novel method where human intervention with the network helped improve the performance of the network
- Human queries give the exact information that the network wants.
- This idea has a future perspective where the author suggest to design a network (generate queries) that would play the role of the user.
- **Please refer the original paper for getting certain insights into the model including the failure cases and semantic analysis of γ vectors.**

Thank You!