

Forest Fire Prediction - Assignment 05

Used technologies

The main programming language this app was developed in was Python, using the Gradio library for the user interface and deploying. Also, we used NumPy and PyTorch (+ TorchVision) for preprocessing and postprocessing operations on the images.

Implementation Steps

After training the models in PyTorch, we imported the model in our application using this function:

```
1 usage  👤 razvan404
def load_model(model_name):
    model_path = os.path.join("binaries", f"{model_name}.pt")
    if not os.path.exists(model_path):
        files = glob.glob(f'saved_models/{model_name}/checkpoint_*.pt')
        path = max(files, key=os.path.getctime)
        save_model_to_binaries(path, model_path)
    return load_binaries(model_path)

model = load_model("ad504ec7-53db-4096-970d-66249dbcf502").to('cuda')
model.eval()
```

We first exported the model into a PyTorch script if not exported for faster inference, then moved it to cuda to use the power of the NVidia GPU to have an even faster inference.

Then, the function of inference which receives a function, preprocess the image, does the inference through the ML model and applies the postprocessing to the obtained segmentation mask is the following one:

```
1 usage  👤 razvan404 *
def predict(image):
    image_transform = build_image_transform(size=(720, 1280))
    transformed_image = image_transform(image)
    image = transformed_image.unsqueeze(0).to('cuda')
    mask = binary_predictions(model(image)).squeeze().cpu()
    return mask.detach().numpy()
```

We resized the images to 720p resolution (1280x720), to test the model inference using a different resolution than the one being trained on (128x128) to prove that, by using only convolutional layers to our model, there is no hard constraint on the resolution of the images.

Lastly, we deployed our application using the following Gradio function call:

```
ui = gradio.Interface(
    fn=predict,
    inputs=gradio.Image(sources=["upload"]),
    outputs="image",
    title="Forest Fire Segmentation",
    allow_flagging="never"
)
ui.launch(share=True)
```

We gave the parameter *share=True* to the launch function to deploy our model to the Gradio servers.

Source code

<https://github.com/razvan404/forest-fire-detection>