

Smart gallery

Matěj Boura
Faculty of Informatics and Management
University of Hradec Kralove,
Hradec Kralove, Czech Republic
bourama1@uhk.cz

Abstrakt—Tato práce se zabývá vývojem inteligentní galerie, využívající umělou inteligenci k automatické klasifikaci fotografií do specifických kategorií. Implementace této galerie by mohla sloužit k vytvoření uživatelsky příjemného prostředí, které umožňuje uživatelům snadno procházet a organizovat svou fotografickou sbírku, primárně u turistických fotografií. Nejdříve bylo nutné prozkoumat techniky hlubokého učení, datasety a metody strojového vidění k efektivnímu a přesnému rozdělování obrázků. Z důvodu vysoké výpočtové náročnosti bylo nutné z datasetu vybrat menší dataset kategorií, které se přímo týkají mého cíle. Výsledkem je model se zhruba o 10% horší přesností, ale 4 násobně rychlejším během predikce.

Klíčová slova—fotografie; zpracování obrazu; klasifikace scény; konvoluční sítě; Tensorflow

I. INTRODUCTION/ÚVOD

A. OBSAH

TODO

II. PROBLEM DEFINITION/ DEFINICE PROBLÉMU

A. OBSAH

TODO

III. NEW SOLUTION / NOVÉ ŘEŠENÍ

A. OBSAH

TODO ... Níže na Obrázku 1 můžete vidět flowchart jak by takováto aplikace mohla fungovat.

IV. IMPLEMENTATION / IMPLEMENTACE ŘEŠENÍ

A. OBSAH

TODO

B. ALGORITHMY

V. TESTING OF DEVELOPED APPLICATION / TESTOVÁNÍ VYVINUTÉ APLIKACE - ŘEŠENÍ

A. OBSAH

TODO

Data: Dataset

Result: Trained Model

```

1 Load the dataset;
2 Split the dataset into training and testing sets;
3 Preprocess the training and testing sets;
4 Create a normalization layer;
5 Build a model with the normalization layer;
6 // Train Model
7 Initialize number of epochs (num_epochs);
8 for epoch = 0 to num_epochs do
9     for each batch in training set do
10         Train the model on the training set;
11         Evaluate the model on the validation set;
12         Compute validation accuracy;
13     end
14 end
15 if validation accuracy does not improve then
16     Create a augmentation layer;
17     Build a model with the addition of augmentation
        layer;
18 end
19 // Train Second Model
20 for epoch = 0 to num_epochs do
21     for each batch in training set do
22         Train the model on the training set;
23         Evaluate the model on the validation set;
24         Compute validation accuracy;
25     end
26 end

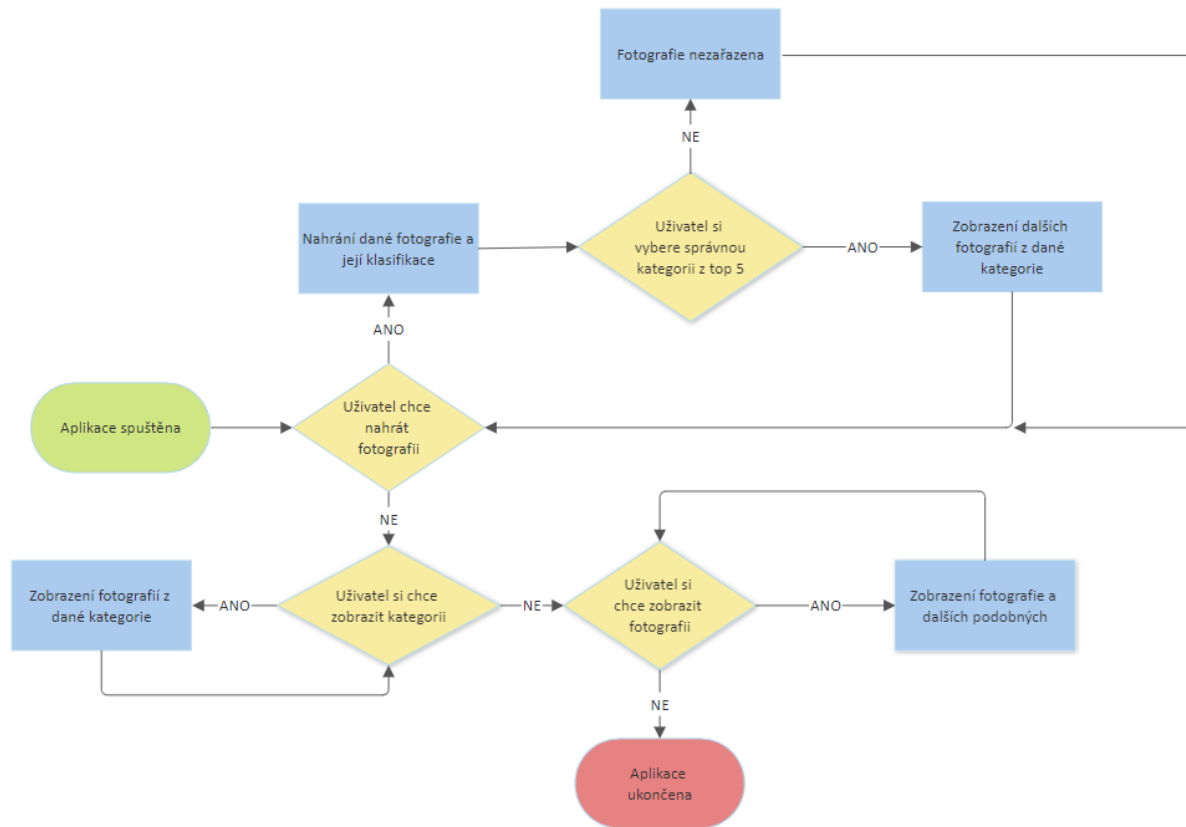
```

Algorithm 1: Create and Train model

VI. CONCLUSIONS / ZÁVĚRY

V rámci této práce byl zkoumán proces vytváření chytré galerie s využitím pokročilé klasifikace obrazů za účelem zlepšení efektivity. Cílem bylo dosáhnout zvýšení rychlosti kategorizace fotografií a současně minimalizovat snížení přesnosti na přijatelnou úroveň. Výsledky práce představují signifikantní zlepšení v oblasti efektivity, kdy bylo dosaženo až čtyřnásobného zvýšení rychlosti klasifikace, a to za cenu snížení přesnosti o zhruba 10%.

Jednou z možných cest vpřed by mohlo být zahrnutí detekce objektu a metody YOLO (You Only Look Once). Tato metoda by mohla efektivně pracovat s různými objekty na jedné fotografii a potenciálně poskytnout přesnější klasifikaci. Vzhledem



Obrázek 1: Flowchart

Data: Dataset, Pretrained Models:

EffNet, MobNet, ResNet, Inception

Result: Best Trained Model

```

1 Load the dataset;
2 Split the dataset into training and testing sets;
3 Preprocess the training and testing sets;
4 for each pretrained model  $M_i$  do
5   Load pretrained model  $M_i$ ;
6   Replace the final classification layer(s) of  $M_i$  with
   a new layer for your task;
7   // Train Modified Model
8   Train the modified model on the training set for
   num_epochs;
9   // Evaluate Model
10  Evaluate the fine-tuned model on the testing set;
11  Compute performance metrics;
12  Store the model and it's performance metrics for
   comparison;
13 end
14 // Select Best Model
15 Select the model with the best performance on the
   testing set;

```

Algorithm 2: Transfer Learning with Pretrained Models

k rychlosti YOLO by mohlo dojít ještě k dalšímu značnému zlepšení výkonu naší galerie.

REFERENCE

- [1] Emmanuel Seetam, Friday E. Onuodu. Object Detection Using Convolutional Neural Network Transfer Learning. 2022, www.semanticscholar.org/paper/Object-Detection-using-Convolutional-Neural-Network-Emmanuel-Onuodu/c61d07cdb15155c9b9fea4af42a3e917f3962a88.
- [2] Al-Shammery, Ali Abbas, et al. "Automatic Image Annotation System Using Deep Learning Method to Analyse Ambiguous Images." Periodicals of Engineering and Natural Sciences (PEN), vol. 11, no. 2, International University of Sarajevo, Apr. 2023, p. 176. <https://doi.org/10.21533/pen.v11i2.3517>.
- [3] Sharma, Tanya Goel Ujjwal. "Real-Time Image Processing Using Deep Learning With Opencv and Python." Journal of Pharmaceutical Negative Results, Medknow, Feb. 2023, pp. 1905–08. <https://doi.org/10.47750/pnr.2023.14.03.246>.
- [4] Wang, Renwu, et al. "Developing an Artificial Intelligence Framework for Online Destination Image Photos Identification." Journal of Destination Marketing and Management, vol. 18, Elsevier BV, Dec. 2020, p. 100512. <https://doi.org/10.1016/j.jdmm.2020.100512>.
- [5] Thakurdesai, Shalva, et al. "Smart Gallery Using Google Vision." Social Science Research Network, RELX Group (Netherlands), Jan. 2021, <https://doi.org/10.2139/ssrn.3852766>.
- [6] Zhou, Bolei, et al. Places: A 10 Million Image Database for Scene Recognition — IEEE Journals & Magazine — IEEE Xplore. 2018, <https://doi.org/10.1109/TPAMI.2017.2723009>.