

## Popis výsledku projektu dokazující dosažení výsledku - Software pro automatickou detekci škůdců na lepových deskách

*Tento dokument popisuje naplnění cíle projektu a postup dosažení výsledku projektu. Data, jednotlivé kódy a funkční algoritmy jsou uloženy u hlavního řešitele projektu.*

Pro sledování vývoje populace polétavých škůdců v porostu se ve sklenících běžně používají kontrolní lepové štítky. Škůdci se na štítku zachytí a jsou následně ručně počítáni, což je pracný a zdoluhavý proces. Lidské počítání je také náchylné k chybám. Software pro automatickou detekci a počítání škůdců na lepových štítcích může tento proces výrazně ulehčit.

Pro účely snadného pořizování snímků lepových štítků přímo ve skleníku a rychlé anotace výskytů škůdců na štítcích byl navržen doporučený postup pro obsluhu. Byly také sestaveny detailní požadavky na mobilní aplikaci, která by měla celý proces usnadnit a urychlit. V plánu bylo také zavedení značení jednotlivých lepových štítků jednoznačným identifikátorem pomocí čárových kódů, které bude možné tisknout přímo ve skleníku. Tato identifikace však nakonec nebyla realizována.

Tato část projektu byla řešena postupně v několika fázích, protože pro trénování modelů umělých neuronových sítí je potřeba zajistit co největší rozmanitost vstupních dat. V tomto případě byly požadovány fotografie štítků pořízené ve všech vegetačních fázích a stádiích porostu a za různých světelných podmínek. Postupem času vzniklo několik různých datasetů a různých klasifikačních a detekčních modelů pro každý dataset.

Na začátku řešení této části projektu nebylo k dispozici dostatečné množství pořízených fotografií, proto byl v prvotní fázi vytvořen uměle vygenerovaný dataset. Nejprve bylo pořízeno několik desítek testovacích fotografií štítků za různých světelných podmínek a úhlů snímání (položené na podložce či pověšené v porostu). Stanovili jsme minimální rozlišení (8 – 12 MPx) a velikost štítku na fotce (min. 75 % fotografie) potřebné pro dostatečně podrobné zachycení menších škůdců, především molic. Ukázka validně pořízené fotografie je na Obr. 1 (a).

Abychom ověřili, že takto pořízená data budou vhodná pro trénování modelu, přistoupili jsme k automatickému vygenerování trénovacích dat odpovídajících zvolenému nastavení focení. Ručně jsme vysegmentovali 70 molic, 50 dalších škůdců a 7 čistých lepových štítků, viz ukázky na Obr. 2. Pomocí algoritmu využívajícího náhody jsme obrázky škůdců rozmístili na jednu ze sedmi fotek čistého štítku. Pro zvýšení variability byl každý obrázek škůdce navíc náhodně zvětšen nebo zmenšen a otočen kolem své osy. Díky tomuto přístupu byla získána anotovaná data. Ke každému obrázku byl uložen xml soubor obsahující informace o třídě a pozici tzv. bounding boxu - bboxu (ohraničujícího rámečku) daného objektu

(škůdce), viz ukázka vygenerované fotky štítku na Obr. 1 (b) a Obr. 3. Tímto způsobem jsme vygenerovali 500 fotek lepových štítků s označenými škůdci.



(a)



(b)

Obr. 1: Lepové štítky se škůdci: (a) foto pořízené ve skleníku Bezdínek, (b) uměle vytvořené foto.

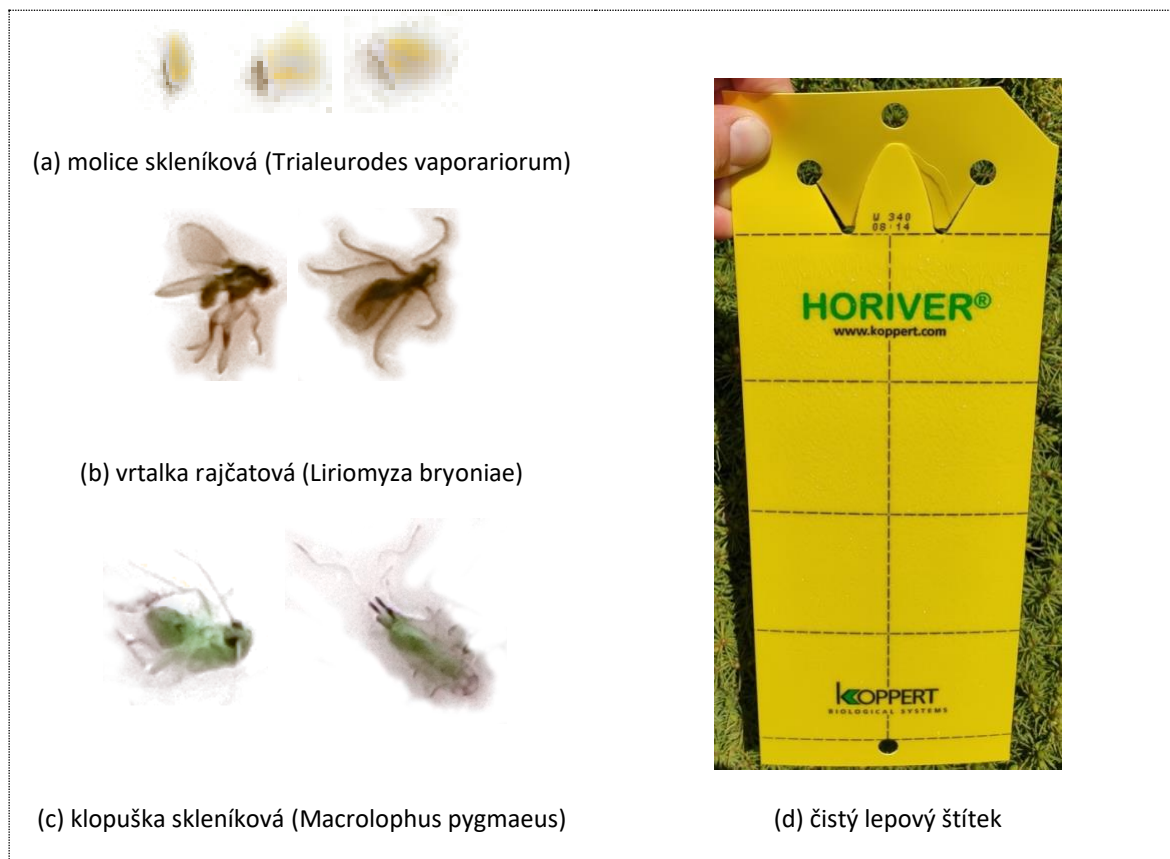
S ohledem na současné trendy mezi publikovanými články a zamýšlené rozšíření na další aplikace jsme se rozhodli pro tvorbu detektoru využívajícího hlubokého učení, konkrétně hluboké konvoluční neuronové sítě. V počítačovém vidění se v současnosti jedná o nejvyužívanější a nejpopulárnější modely.

Pro prvotní testy byla vybrána konvoluční síť typu Faster-R-CNN<sup>1</sup>, která je často používána i dalšími výzkumníky zabývajícími se detekcí v oblasti skleníků a zemědělství. Abychom síti zaručili dostatečně detailní pohled na lepový štítek, byla nejdříve vstupní fotografie rozdělena na segmenty o velikosti 800 x 1024 pixelů. Detekce se provádí pro každou ořezanou část vstupní fotografie zvlášť. Konečný výsledek je tvořen složením jednotlivých detekovaných částí.

Vygenerovaná data byla rozdělena na trénovací a validační množinu v poměru 9:1. Pro zvýšení variability dat během trénování byl vždy vybrán náhodný výřez z fotografie v trénovací množině a aplikování augmentace dat zrcadlením. Při trénování byla síť inicializována předtrénovanými váhami. Model byl

<sup>1</sup> Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

trénovaný po dobu 10 epoch, přičemž za jednu epochu se považuje průchod všemi trénovacími obrázky. Pro úpravu vah během trénování byl využit optimalizační algoritmus SGD (Stochastic gradient descent) s následujícími parametry: learning rate 0.005, momentum 0.9 a weight\_decay 0.0005.



Obr. 2 Dílčí obrázky použité pro generování trénovacích dat: (a) molice skleníková, (b) vrtalka rajčatová, (c) klopůška skleníková, (d) čistý lepový štítek.

Dataset byl následně rozšířen o testovací množinu s anotovaným počtem výskytu Molice bavlníkové na každé fotografii kontrolního lepového štítku, aby mohlo být provedeno vyhodnocení úspěšnosti modelu natrénovaného nad datasetem vygenerovaných fotografií. Anotace počtu molic byla provedena personálem ve skleníku pohledem na lepovou desku a rostlinolékařem pohledem na fotografii dané lepové desky, jehož počet byl určen jako referenční hodnota. Vyhodnocení úspěšnosti poslední verze detekčního modelu má hodnotu Precision 0.78, Recall 0.87 a F1-skóre 0.82, viz Tab. 1.

Tab.1: Vyhodnocení výsledků systému pro detekci molic na lepových štítcích ve srovnání s počty stanovenými obsluhou skleníku a rostlinolékařem

	Počet molic	TP	FP	FN	Precision	Recall	F1-skóre
Rostlinolékař	185	–	–	–	–	–	–
Obsluha	265	184	81	1	0.69	0.99	0.81
Navržený systém	207	160	46	24	0.78	0.87	0.82

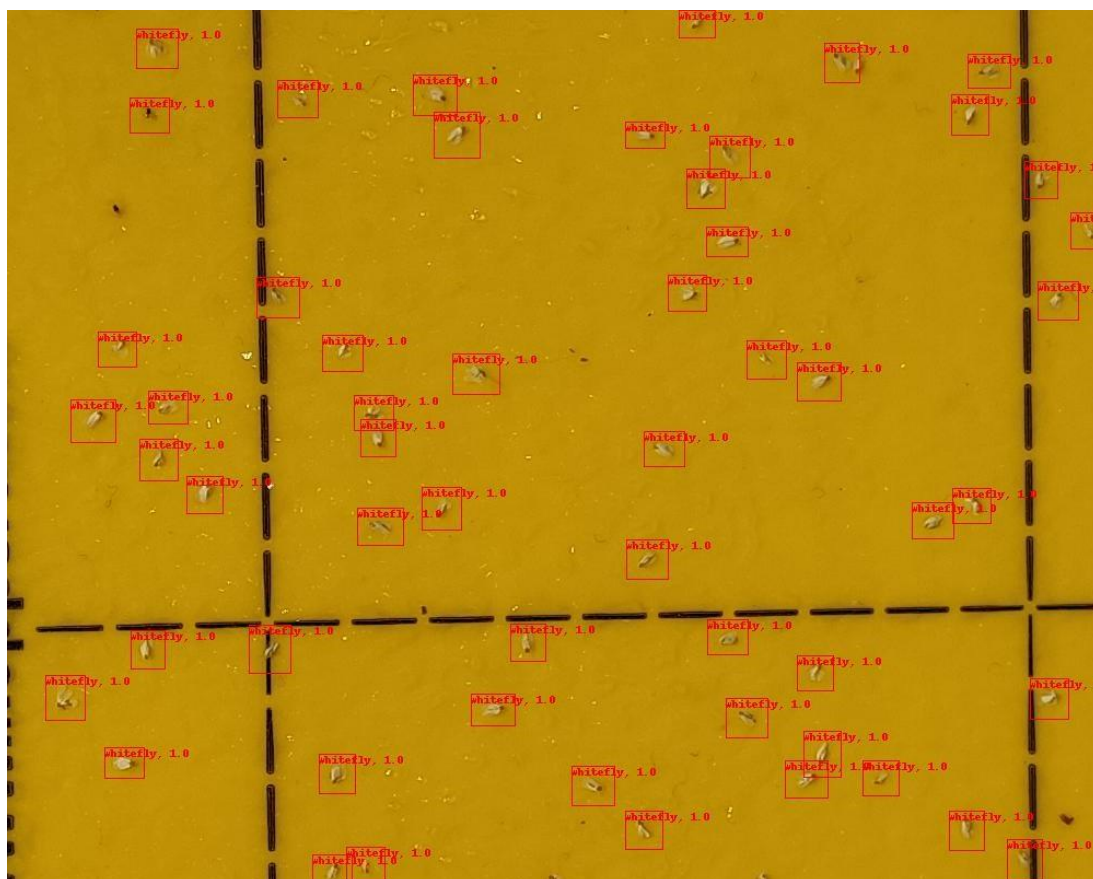
Tab.2: Vyhodnocení výsledků systému pro detekci molic na lepových štítcích na testovací části uměle generovaného datasetu.

Škůdce	Počet bboxů	Detekováno	Recall	AP
vrtalka	12059	12500	0.820	0.817
klopuška	17	21	0.832	0.823
molice	1180	2201	0.790	0.784
mAP	-	-	-	0.812

Další dataset vznikl úpravou testovací množiny obrázků tak, že výskyt škůdců v jednotlivých fotografiích byl anotován bounding boxy (ohraničujícími rámečky, obr. 4). Jako anotační software byl zvolen nástroj Labellmg a anotace byla provedena rostlinolékařem. Tato úprava umožnila vyhodnocování detekčních výsledků metrikami beroucími v potaz míru překrytí detekovaného a originálního bounding boxu. Běžně se k tomuto využívá hodnota 0.5 IOU (intersection over union), kdy se dotýčné dva bounding boxy překrývají alespoň polovinou své společné plochy. Při této analýze se ukázalo, že model natrénovaný na uměle vygenerovaných datech není schopen detekční účinnost přenést i na detekci ve snímcích reálně pořízených ve skleníku. Hodnoty Average Precision a Average Recall při IOU 0.5 testované na anotovaných snímcích klesly pod 0.5. Dostupné snímky navíc byly pořízeny během poměrně krátkého časového úseku a neobsahovaly dostatek výskytů škůdců pro kvalitní natrénování hluboké konvoluční sítě. Dataset byl proto dále postupně rozšiřován anotacemi nových fotografií, které prováděl pomocný personál. Rostlinolékař anotace pouze validoval či dodatečně editoval před zařazením do datasetu. Výsledný dataset obsahuje celkem 94 obrázků lepových štítků a 1752 označených molic.

Postup trénování a predikce byl zachován obdobně jako v případě generovaného datasetu. Vstupní snímky byly rozděleny na trénovací (60 snímků), validační (15 snímků) a testovací množinu (19 snímků). Pro trénování a validaci byly snímky rozkrájeny na dlaždice o velikosti 512x512 pixelů, díky překryvu výřezů tak dataset obsahuje finálně 8038 obrázků a 3057 anotovaných molic. Výsledky validační množiny byly hodnoceny na těchto výřezích. Výsledky testovací množiny pak byly vypočteny vždy až pro celý obrázek lepového štítku a to tak, že predikce z jednotlivých výřezů byly sloučeny. Překrývající se detekce na rozhraních částí obrázku byly sloučeny, pokud jejich překryv IoS (Intersection over smaller area) byl větší než 0.6. Obr. 4. ukazuje příklad detekce molic na výřezu snímku lepového štítku z validační množiny.

Pro postupnou detekci každého snímku v testovací množině po jednotlivých výřezích natrénovaným modelem a zpětné spojení jednotlivých detekcí škůdců do celkového snímku (ukázka výsledků detekce na obrazu celého lepového štítku je na obr. 4) byla použita knihovna SAHI<sup>2</sup>.



Obr. 3: Příklad detekce škůdců ve výřezu snímku žlutého lepového štítku z validační množiny

<sup>2</sup> F. C. Akyon, S. Onur Altinuc and A. Temizel, "Slicing Aided Hyper Inference and Fine-Tuning for Small Object Detection," 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 2022, pp. 966-970, doi: 10.1109/ICIP46576.2022.9897990.





Obr. 4: Příklad detekce škůdců na snímku žlutého lepeného štítku z testovací množiny

Výsledné řešení dosahuje na testovací množině Precision 0.8378, Recall 0.8254 a F1-skóre 0.8314 (jedná se o harmonický průměr mezi recall a precision). Jsou to obdobné hodnoty, jako v prvotním testování dosahovala obsluha vs rostlinolékař 0.081 (viz Tab. 1). Oproti původně navrženému systému však započítáváme opravdu jen správné detekce, tedy ty, které se překrývají s původním anotovaným bounding boxem hodnotou 0.5 IoU (intersection over union). Tato přesnost detekce byla shledána dostatečnou pro reálné použití ve skleníku.

Výsledný TOODr50 model byl také implementován v softwarovém balíčku s rozhraním API zprostředkávající detekci molic na snímku žluté lepené desky. Software obsahuje funkci *inference\_engine*, která se stará o inicializaci modelu, který je popsán konfiguračním *.py* souborem dle knihovny MMDetection<sup>3</sup> a natrénovanými váhami uloženými ve formátu *.pth*. V případě potřeby je možné model jednoduše zaměnit za jiný. Dalším povinným vstupem je cesta k vstupnímu obrázku, případně může být i zadána jen cesta k adresáři obsahujícímu více obrázků, v takovém případě budou predikovány všechna obrazová data ve složce. Podporované formáty vstupních obrázků jsou následující: JPG, JPEG, BMP, PNG. Posledním vstupem je cesta k adresáři, kde jsou po spuštění programu uloženy výsledky predikce.

Program pracuje ve třech krocích. V první fázi dojde k inicializaci modelu a jeho přípravě pro predikci. V případě potřeby je možné upravit podrobnější podmínky predikce, například vybrat, zda výpočty budou probíhat na CPU či GPU. Dále je možné volitelně uložit i vizualizaci výstupu. V tom případě bude ve výstupní složce vytvořen adresář *visuals* obsahující všechna vstupní obrazová data s vykreslenými

<sup>3</sup> Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., ... & Lin, D. (2019). MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*.

detekcemi. Po inicializaci modelu program načte vstupní obrazová data a vytvoří JSON soubor *test.json*, kde jsou vstupní obrázky uloženy v COCO<sup>4</sup> formátu. Poté probíhá samotná inference, během které je vstupní obraz rozřezán na dílčí části, na kterých je provedena predikce. Výsledné detekce jsou následně vyfiltrovány a opět složeny dohromady.

Výsledky detekce jsou uloženy do JSON souboru ve standardním COCO<sup>5</sup> formátu, kde jsou k dispozici pro případné další zpracování. JSON soubor obsahuje především seznam obrázků s přiděleným *IMAGE\_ID*, dále seznam dostupných tříd s přiděleným *CATEGORY\_ID* a konečně seznam anotací s přiděleným vlastním ID pomocí *IMAGE\_ID* a *CATEGORY\_ID* navázaných ke kategorii a jednotlivým obrázkům. Je tedy možné například jednoduše zjistit počet molic detekovaných na lepovém štítku.

---

<sup>4</sup> Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer, Cham.

<sup>5</sup> Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer, Cham.