Vysoké učení technické v Brně Ministerstvo vnitra ČR

Fakulta informačních technologií

***FACIS: Sada forenzních analytických nástrojů ke zpracování obrazu a videa pro službu kriminální policie a vyšetřování***

**Identifikační kód *VJ020100041***

**Název předkládaného výsledku:**

***Extractor***

|  |  |  |
| --- | --- | --- |
| **Typ výsledku dle UV č. 837/2017** | **Evidenční číslo (příjemce)** | **Rok vzniku** |
| ***R***  software | R2 | 2023 |
| **ISBN-ISSN** | **Webový odkaz na výsledek** | **Kde a kdy publikováno** |
|  | https://www.fit.vut.cz/research/product/805/ | web |

**Stručná anotace k výsledku:**

Nástroj Extractor je pokročilé řešení pro extrahování multimediálních dat z adresářů a souborů. Identifikuje obrázky a videa ukrytá v různých formátech, souborech a adresářích. Mezi další významné funkce se řadí transformace obrázků do jednotného formátu PNG a zachování videí jako klíčových snímků, což usnadňuje následné zpracování. Extractor navíc automaticky generuje miniatury pro každý obrázek.

Extractor podporuje přes 60 multimediálních formátů, včetně nejčastějších formátů, jakými jsou BMP, JPG, PNG, WEBM, MP4, a MPEG formáty. Extractor rozpoznává bezmála 100 různých formátů pro detekci multimediálního obsahu, kromě již zmíněných i formáty dokumentů (PDF, DOC, XLS,...) a archivů (RAR, ZIP,...).

**Řešitelský tým:**

*Dominika Regéciová, Dušan Kolář, Jan Pluskal, Ondřej Ryšavý*

Obsah

[Úvod 1](#_Toc153909921)

[Technická dokumentace 1](#_Toc153909922)

[Instalace 1](#_Toc153909923)

[Spuštění aplikace 3](#_Toc153909924)

[Uživatelský manuál 4](#_Toc153909925)

[Runtime aplikace 4](#_Toc153909926)

[Programátorská dokumentace 6](#_Toc153909927)

[Analýza datových formátů 7](#_Toc153909928)

[Závěr 9](#_Toc153909929)

## Úvod

Tento projekt si klade za cíl vytvořit kolekci nástrojů, která je schopna rychle a efektivně detekovat a extrahovat multimediální obsah z obrazů pevných disků. Primárním cílem této implementace je umožnit automatizovaný způsob extrakce zájmových souborů. Extractor je primárně nástroj, který umožňuje detekovat požadovaný multimediální obsah, včetně prohledávání záměrně schovaného obsahu v rámci jiných formátů a skrytých složek. Mezi další užitečné úlohy Extractoru se řadí i schopnost převádět nalezený obsah na jednotný PNG formát, vytváření miniatur pro vizualizaci a u videí pak také získávání klíčových snímků, které jsou nápomocné pro další zpracování (např. detekce postav).

## Technická dokumentace

Technická dokumentace je rozprostřena do sekcí Instalace, Uživatelský manuál, Programátorská dokumentace, Analýza datových formátů, a Nápomocné skripty.

## Instalace

Nástroj je balíčkován v Nix [[1]](#footnote-1)správci balíčků. Tento přístup umožňuje mít plnou kontrolu nad závislostmi nástroje, který se tímto stane naprosto nezávislý na operačním systému. Nástroj je možné tedy provozovat na strojích s architekturou amd64 jako program, vytvořit LXC/Docker/Podman kontejner či specializovaný virtuální stroj.

Nativní instalace se provádí importem Extractor flake modulu a přidáním balíčku extractor.packages."x86\_64-linux".extractor mezi systémové balíčky.

Alternativně, případně pro vývoj je možné naklonovat repositář projektu a následně vstoupit do vývojářského shell.

nix develop

Ve vývojářském shellu jsou připraveny všechny závislosti projektu dle deklarace modulu Extractoru[[2]](#footnote-2).

Prostředí Nix je možné použít i k sestavení projektu pomocí:

čímž vznikne výsledek dostupný jako spustitelný skript ./result/bin/extractor.sh ve složce ./result.

nix build .#extractor

V případě potřeby je možné sestavit spustitelný nástroj i čistě pomocí standardních kontejnerizačních nástrojů jako je Docker, nebo Podman a provozovat nástroj Extractor v kontejneru.

[podman|docker] build . -t facis/ extractor:latest

### Spuštění aplikace

Pokud je nástroj Extractor využíván v NixOS a je tedy importován pomocí flake, stává se systémovým nástrojem a je možné ho spustit jako jiný libovolný systémový nástroj:

./result/bin/extractor.sh

Stejným způsobem lze Extractor zpustit i pokud je nástroj sestaven ze zdrojových kódů získaných z repositáře projektu a přeložen pomocí Nix balíčkovacího správce.

V případě, že je nástroj Extractor sestaven jako kontejner, spuštění je následovné:

[podman|docker] run --rm facis/extractor:latest

### Argumenty

Extractor očekává 3 vstupní informace: režim pojmenování vzniklých souborů, vstupní a výstupní složky a seznam souborů, které mají být zkontrolovány.

Extractor má dva režimy pojmenování, buď se vypočítá sha256 hash z obsahu souboru, nebo je generováno unikátní GUID (toto pojmenování je unikátní pro každý běh programu, zatímco hash zůstává stejný). Režim je možné volit pomocí argumentů -guid, a nebo -sha.

Extractor dále potřebuje údaje, kde nalezne vstupní soubory (cesta k složce se soubory) a cestu ke složce, kam má vkládat výsledky.

Jako poslední informace se předává seznam souborů, které má Extractor zpracovat. Zde jsou opět dva způsoby, jak předat tuto informaci. Buď lze použít argument -csv a předat Extractoru cestu k CSV souborů obsahující seznam souborů. Druhým způsobem je vstup přes JSON formát předaný jako vstupný řetězec.

./extractor.sh -sha -csv input\_dir output\_dir csv\_file

nebo

./extractor.sh -guid -json input\_dir output\_dir json\_text

## Uživatelský manuál

Tato aplikace je primárně určena pro systémové integrátory, kteří automatizují extrakci multimediálních dat z obrazů disků. Aktuální kapitola stručně vysvětlí proces běhu aplikace a načítání rozšíření.

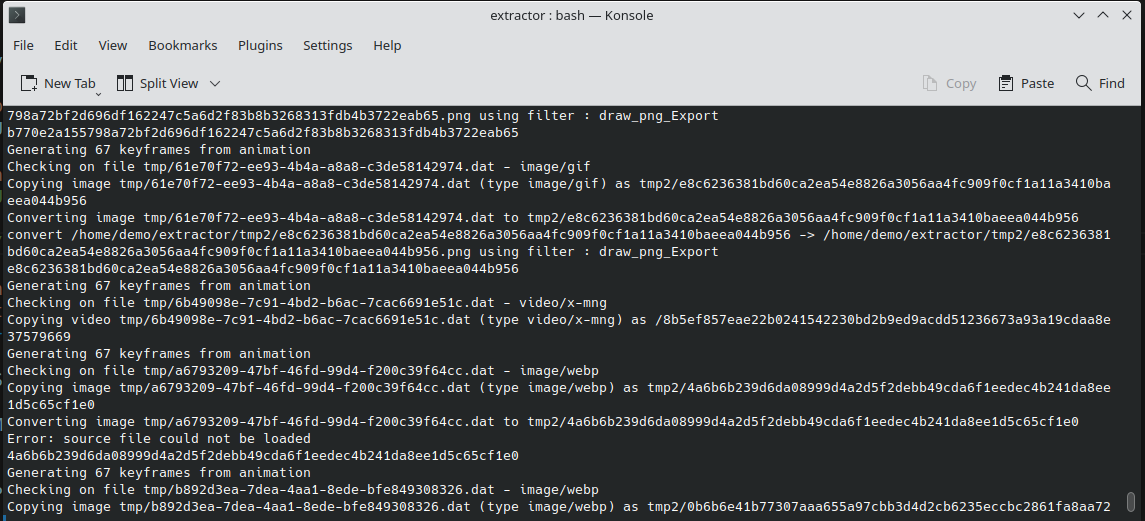
### Runtime aplikace

Extractor zpracovává seznam souborů, které jsou definované buď v rámci CSV souboru, nebo v rámci JSON vstupu.

CSV formát data o vstupních souborech v podobě: OriginalPath, TargetPath, FileSize, Width, Height, FileTypeInfo, Md5Hash, ShaHash. Extractor postupně načítá soubor za souborem a zpracovává jej. Pokud se jedná o soubor, ve kterém mohou být další soubory (archiv, dokument), Extractor zkoumá i tento obsah a vytváří tak kopie nových souborů.

V případě, že je vstup JSON formát, pak je definován jako '{"FilePath": "path", "Id": "id"}'. V tomto režimu Extractor přijímá v rámci jednoho běhu pouze 1 vstupní soubor.

Extractor poskytuje podrobné informace o zpracovávaných souborech pomocí logů, jejichž ukázka je vidět na obrázku 1.



Obrázek 1 Ukázka běhu nástroje Extractor extrahující multimediální data.

Výsledek extrakce zájmových dat je pak k dispozici v adresářové struktuře, kde jsou nalezené obrázky ve formátu PNG, miniatury rovněž v PNG formátu a videa. Videa a animace rovněž generují složky, které obsahují klíčové snímky a jejich miniatury, jak je vidět na obrázku 4. Součástí jsou také dva výstupní CSV soubory: extractor\_full.csv obsahující všechny zpracované soubory a informace o nich a extartor.csv, který obsahuje seznam obrázkových souborů pro další kroky zpracování, například pro detekci postav. Ukázka tohoto výstupu je k dispozici na obrázku 2.

OriginalPath,TargetPath,FileSize,Width,Height,FileTypeInfo,Md5Hash,ShaHash,Thumbnail

extractor\_test\_data/animations/apng/butterfly\_apng.png,b770e2a155798a72bf2d696df162247c5a6d2f83b8b3268313fdb4b3722eab65,2438998,250,250,image/png,181c02697959c959389d2948596840f1,b770e2a155798a72bf2d696df162247c5a6d2f83b8b3268313fdb4b3722eab65,b770e2a155798a72bf2d696df162247c5a6d2f83b8b3268313fdb4b3722eab65\_thumb.png

extractor\_test\_data/animations/apng/butterfly\_apng.png/00.png,ada565e08eee5935b243c9c2bf16d52e1e57bc51d2ca75b1594e47c1f0107b66,42176,250,250,image/png,4eb31dd72624fac70e8f655057b6d04d,ada565e08eee5935b243c9c2bf16d52e1e57bc51d2ca75b1594e47c1f0107b66,173d6d1d67bd513166a03385e4a8981717eedee91c90422d440edfd33e0a96d4

Obrázek 2 Ukázka logu nástroje Extractor.

V případě, že byl použit JSON režim, je na výstupu rovněž generován zápis v JSON formátu v záznamech oddělující každý soubor, jak je vidět na obrázku 3. V tomto režimu také nejsou produkovány miniatury.

Obrázek 3 Ukázka JSON výstupu z Extractoru.

./extractor.sh -sha -json images tmp3 '{"FilePath": "images/jpg/white\_cat\_jpg.jpg", "Id": "white\_cat14"}'

Výstup je následovný:

{"OriginalPath":"white\_cat14", "TargetPath": "5b78de6222844c863ac089c79c4d5cdb1226bb41fbca777e303d7f5fb3aef5ee.png","FileSize": "40531","Width": "728","Height": "712","FileTypeInfo": "image/jpeg","Md5Hash": "46ec6ea8275d6943e72dbac7a0bfc1cd","ShaHash": "5b78de6222844c863ac089c79c4d5cdb1226bb41fbca777e303d7f5fb3aef5ee"}

## 

Obrázek 4 Ukázka struktury výstupu nástroje Extractor.

## Programátorská dokumentace

Jádrem projektu Extractoru je bashovský skript extractor.sh[[3]](#footnote-3). Ten zajišťuje procházení všech souborů a předává práci dalším nástrojům. Další důležitou součástí projektu je skript napsaný v jazyce Python, extractor.py[[4]](#footnote-4). Tento skript splňuje hned několik funkcionalit – extrahuje soubory vložené do pdf dokumentů, extrakce animovaných souborů na snímky a vytváření miniatur. Závislosti pro tento skript jsou řešené přes Poetry[[5]](#footnote-5), manager pro balíčkování správu závislostí.

Závislosti pro externí nástroje jsou popsány v rámci flake souboru[[6]](#footnote-6). Jedná se především o nástroj ffmpeg pro zpracování videí, libreoffice pro konverzi dokumentů a obrázků a nástroje pro rozbalování a dekompresi archivů (např. unzip, unrar,…).

Pro zajištění konzistentního deklarativně konfigurovaného prostředí je k dispozici Nix modul definující veškeré závislosti. Zpřístupnění vývojového shell je pak možné následovně:

nix develop

Následný vývoj je pak možné provádět s využitím libovolného IDE. Pro vývoj bylo použité prostředí Visual Studio Code.

Kód dodržuje zásady CleanCode[[7]](#footnote-7) a je tedy dle zásad CleanCode samodokumentující.

Projekt také obsahuje regresivní testy pro otestování funkcionality důležité především během vývoje. Skript test\_extractor.sh zajišťuje stažení testovacího datasetu, spuštění testu a ověření výsledků. Lze ho spustit se zadáním cesty pro hlavní extractor.sh skript:

./test\_facis.sh .

## Analýza datových formátů

Na světě je nespočet datových souborů a není možné je podporovat všechny, včetně neobvyklých a velmi zřídka používaných formátů. Přesto byl cíl pokrýt co největší množinu těchto formátů s důrazem na časté a běžné formáty. Pro možnost evaluovat pokrytí těchto formátů byla vytvořena studie datových formátů. S pomocí firmy Gen Digital [[8]](#footnote-8)byly získány informace z databáze Virus Total[[9]](#footnote-9), která obsahovala informace o souborech nahraných pro kontrolu proti malwaru za posledních 90 dní.

Přes Virus Total denně prochází miliony souborů různých formátů, získali jsme tedy dostatečně reprezentativní vzorek s širokým zastoupením různých multimediálních, ale i dalších formátů. Pro naše potřeby jsme zkoumali kategorie obrázky, videa (a animace), dokumenty a archivy. Z počtu nahraných souborů pak byla vypočteno percentuálně zastoupení formátu v dané kategorii. Pokud jsme si byly vědomi formátu, který se v datasetu nevyskytl, byla mu přidána normalizační hodnota 0.001 %. Pokrytí všech souborů pak značí 100 %, je ale nutno dodat, že tato hodnota je spíše ilustrativní, protože jak jsme psali již výše, není možné podporovat všechny možné formáty, které se na světě kdy objevily.

Záměrem je však snaha o maximální přiblížení této hodnotě a zajistit podporu pro soubory, které je možné během provozu očekávat. V rámci obrázků uvádíme nejčastější formáty u obrázků a videí (bez kompletního výčtu). Dle našich výpočtů, v obou kategoriích je pokryto více jak 94 % formátů, přes 60 souborů. Pokud počítáme i formáty mimo multimediální obsah, který je však nutné kontrolovat, Extractor rozpoznává a detekuje přes 100 formátů. Do budoucna budou tyto výsledky ještě lepší, jak bude přidávána podpora pro méně časté formáty.

Obrázek 5 U obrázků byly nejčastější formáty BMP, JPG a PNG. Extractor nyní dosahuje 94% pokrytí.

Obrázek 6 U videí byly nejčastější formáty WEBM, MP4 a SWF. Extractor dosahuje pokrytí 98%.

## Nápomocné skripty

Repositář Extractoru obsahuje přídavné pomocné skripty, které zpříjemňují nejen vývoj, ale také použití pro uživatele. První je nástroj simulate\_maxtor.sh[[10]](#footnote-10). Maxtor[[11]](#footnote-11) je prvním nástrojem prezentovaným v rámci FACIS projektu, s tím, že zpracovává data z obrazů disků a vytváří seznam souborů, které má následně Extractor zpracovat. Protože však vstupem může být i čistě složka, byl vytvořen tento pomocný skript, který simuluje činnost Maxtoru (ve velmi zjednodušené podobě) a vytváří CSV soubor, který lze pak použít pro Extractor. Spouští se s informací o vstupní a výstupní složce a výstupním CSV souboru:

./simulate\_maxtor.sh input\_dir output\_dir csv\_file

Druhý skript slouží pro ještě příjemnější použití Extractoru a spojuje krok simulace Maxtoru a spouštění Extractoru. Skript demo\_facis.sh [[12]](#footnote-12)pak produkuje výsledky v CSV režimu. Na vstupu skript potřebuje pouze údaje o vstupní a výstupní složce:

./demo\_facis.sh input\_dir output\_dir

## Závěr

Nástroj Extractor je nepostradatelným přínosem pro ty, kteří hledají rafinované řešení pro extrakci multimediálních dat. Jeho schopnost plynule procházet různými formáty a jeho závazek ke standardizaci z něj činí cenný doplněk jakékoli sady nástrojů pro zpracování dat.

1. https://nixos.org/learn [↑](#footnote-ref-1)
2. https://gitlab.nesad.fit.vutbr.cz/facis/extractor/-/blob/master/flake.nix [↑](#footnote-ref-2)
3. https://gitlab.nesad.fit.vutbr.cz/facis/extractor/-/blob/master/extractor.sh [↑](#footnote-ref-3)
4. https://gitlab.nesad.fit.vutbr.cz/facis/extractor/-/blob/master/extractor.py [↑](#footnote-ref-4)
5. https://python-poetry.org/ [↑](#footnote-ref-5)
6. https://gitlab.nesad.fit.vutbr.cz/facis/extractor/-/blob/master/flake.nix [↑](#footnote-ref-6)
7. https://www.amazon.com/Clean-Code-Handbook-Software-Craftsmanship/dp/0132350882 [↑](#footnote-ref-7)
8. https://www.gendigital.com [↑](#footnote-ref-8)
9. https://www.virustotal.com [↑](#footnote-ref-9)
10. https://gitlab.nesad.fit.vutbr.cz/facis/extractor/-/blob/master/simulate\_maxtor.sh [↑](#footnote-ref-10)
11. https://www.fit.vut.cz/research/product/748/ [↑](#footnote-ref-11)
12. https://gitlab.nesad.fit.vutbr.cz/facis/extractor/-/blob/master/demo\_facis.sh [↑](#footnote-ref-12)