



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У
НОВОМ САДУ



Немања Малиновић

Развој OCR модула за форензички алат Autopsy

ЗАВРШНИ РАД

- Мастер академске студије -

Нови Сад, 2025



КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, **РБР**:

Идентификациони број, **ИБР**:

Тип документације, **ТД**:

монографска документација

Тип записа, **ТЗ**:

текстуални штампани материјал

Врста рада, **ВР**:

мастер рад

Аутор, **АУ**:

Немања Малиновић

Ментор, **МН**:

Стеван Гостојић

Наслов рада, **НР**:

Развој OCR модула за форензички алат Autopsy

Језик публикације, **ЈП**:

српски

Језик извода, **ЈИ**:

српски

Земља публиковања, **ЗП**:

Република Србија

Уже географско подручје, **УГП**:

Аутономна покрајина Војводина

Година, **ГО**:

2025

Издавач, **ИЗ**:

ауторски репринт

Место и адреса, **МА**:

Нови Сад; Трг Доситеја Обрадовића 6

Физички опис рада, **ФО**:

(поглавља/страна/ цитата/табела/слика/графика/прилога)

7/62/0/8/22/0/0

Научна област, **НО**:

електротехничко и рачунарско инжењерство

Научна дисциплина, **НД**:

дигитална форензика

Предметна одредница/Кључне речи, **ПО**:

OCR, индексирање, препроцесирање, Autopsy, модул

УДК

Чува се, **ЧУ**:

Библиотека Факултета техничких наука, Нови Сад

Важна напомена, **ВН**:

Извод, **ИЗ**:

У оквиру савремене дигиталне форензике, обрада и анализа мултимедијалних садржаја представљају кључни корак у процесу истраживања дигиталних трагова. Овај рад се бави развојем модула за софтвер Autopsy, који омогућава примену технологије оптичког препознавања карактера (OCR) над сликама. За реализацију је коришћен Tesseract OCR у комбинацији са алатима за предобраду слика, како би се побољшала тачност препознавања текста. Развијен модул аутоматизује процес екстракције текста, његово индексирање и повезивање са постојећим подацима у оквиру форензичке истраге. Рад доприноси унапређењу Autopsy окружења.

Датум прихватања теме, **ДП**:

Датум одбране, **ДО**:

Чланови комисије, **КО**:

Председник:

др Милан Делић, ред. проф.

Члан:

др Марко Марковић, ванр. проф.

Члан:

Члан, ментор:

др Стеван Гостојић, ред. проф.

Потпис ментора



KEY WORDS DOCUMENTATION

Accession number, **ANO**:

Identification number, **INO**:

Document type, **DT**: monographic publication

Type of record, **TR**: textual printed material

Contents code, **CC**: master thesis

Author, **AU**: Nemanja Malinović

Mentor, **MN**: Stevan Gostojić

Title, **TI**: Developing OCR Module for Autopsy Forensics Tool

Language of text, **LT**: Serbian

Language of abstract, **LA**: Serbian

Country of publication, **CP**: Republic of Serbia

Locality of publication, **LP**: Autonomous Province of Vojvodina

Publication year, **PY**: 2025

Publisher, **PB**: author's reprint

Publication place, **PP**: Novi Sad, Dositeja Obradovica sq. 6

Physical description, **PD**: 7/62/0/8/22/0/0

(chapters/pages/ref./tables/pictures/graphs/appendixes)

Scientific field, **SF**: electrical and computer engineering

Scientific discipline, **SD**: digital forensics

Subject/Key words, **S/KW**: OCR, indexing, preprocessing, Autopsy, module

UC

Holding data, **HD**: The Library of Faculty of Technical Sciences, Novi Sad, Serbia

Note, **N**:

Abstract, **AB**:

With modern digital forensics, the processing and analysis of multimedia content represents a key step in the investigation of digital traces. This thesis focuses on the development of a module for the Autopsy software, which enables the application of Optical Character Recognition (OCR) technology on images. For the implementation, Tesseract OCR was used in combination with image preprocessing tools in order to improve text recognition accuracy. The developed module automates the process of text extraction, its indexing, and integration with existing data within the scope of a forensics investigation. The thesis contributes to the enhancement of the Autopsy environment.

Accepted by the Scientific Board on, **ASB**:

Defended on, **DE**:


Defended Board, **DB**: President: dr Milan DeliĆ, full prof.

Member: dr Marko Marković, assoc. prof.

Member:

Member, Mentor: dr Stevan Gostojić, full prof.

Menthor's sign

	УНИВЕРЗИТЕТ У НОВОМ САДУ • ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА 21000 НОВИ САД, Трг Доситеја Обрадовића 6	Број:
	ЗАДАТАК ЗА ЗАВРШНИ РАД	Датум:

(Податке уноси предметни наставник - ментор)

Студијски програм:	Рачунарство и аутоматика		
Студент:	Немања Малиновић	Број индекса:	E2 45/2023
Степен и врста студија:	мастер академске студије		
Област:	електротехничко и рачунарско инжењерство		
Ментор:	др Стеван Гостојић, ред. проф.		
НА ОСНОВУ ПОДНЕТЕ ПРИЈАВЕ, ПРИЛОЖЕНЕ ДОКУМЕНТАЦИЈЕ И ОДРЕДБИ СТАТУТА ФАКУЛТЕТА ИЗДАЈЕ СЕ ЗАДАТАК ЗА ЗАВРШНИ РАД, СА СЛЕДЕЋИМ ЕЛЕМЕНТИМА: – проблем – тема рада; – начин решавања проблема и начин практичне провере резултата рада, ако је таква провера неопходна;			

НАСЛОВ ЗАВРШНОГ РАДА:

Развој OCR модула за форензички алат Autopsy

ТЕКСТ ЗАДАТКА:

<ol style="list-style-type: none"> 1. Анализирати стање у области оптичког препознавања карактера (OCR) и развоја модула за форензички алат Autopsy. 2. Специфицирати захтеве OCR модула за форензички алат Autopsy. 3. Специфицирати дизајн модула на основу специфицираних захтева. 4. Имплементирати специфициран модул. 5. Тестирати имплементиран модул. 6. Документовати (1), (2), (3), (4) и (5).
--

Руководилац студијског програма:	Ментор рада:

Примерак за: <input type="checkbox"/> - Студента; <input type="checkbox"/> - Ментора
--

Образац Q2.HA.04-03 - Издање 1



УНИВЕРЗИТЕТ У НОВОМ САДУ • ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА
21000 НОВИ САД, Трг Доситеја Обрадовића 6

ИЗЈАВА О НЕПОСТОЈАЊУ СУКОБА ИНТЕРЕСА

Изјављујем да нисам у сукобу интереса у односу ментор – кандидат и да нисам члан породице (супружник или ванбрачни партнер, родитељ или усвојитељ, дете или усвојеник), повезано лице (крвни сродник ментора/кандидата у правој линији, односно у побочној линији закључно са другим степеном сродства, као ни физичко лице које се према другим основама и околностима може оправдано сматрати интересно повезаним са ментором или кандидатом), односно да нисам зависан/на од ментора/кандидата, да не постоје околности које би могле да утичу на моју непристрасност, нити да стичем било какве користи или погодности за себе или друго лице било позитивним или негативним исходом, као и да немам приватни интерес који утиче, може да утиче или изгледа као да утиче на однос ментор-кандидат.

У Новом Саду, дана 08.10.2025. _____

Ментор

Кандидат

ПРЕГЛЕД САДРЖАЈА

Преглед садржаја	xi
Списак слика	xiii
Списак табела	xv
Списак листинга	xvii
Захвалница	xix
Увод	1
Стање у области.....	3
Оптичко препознавање карактера и индексирање текста.....	17
Спецификација.....	31
Имплементација	39
Демонстрација.....	45
Закључак.....	57
Библиографија.....	59
Биографија.....	62

СПИСАК СЛИКА

<i>Број</i>	<i>Страна</i>
1. „Чаробњак“ за креирање новог случаја	5
2. Типови извора података	6
3. Опција Logical files	8
4. Одабир Ingest модула	8
5. Дијаграм случајева коришћења	32
6. Класни дијаграм	36
7. Дијаграм секвенци	38
8. Кориснички интерфејс при покретању модула	45
9. Опције избора језика	47
10. Ingest Inbox	47
11. Приказ логова система.....	48
12. Жути билборд	49
13. Резултат анализе – жути билборд.....	49
14. Ћирилични текст	50
15. Резултат анализе – Ћирилични текст.....	50
16. Рукопис.....	52
17. Резултат анализе – Рукопис.....	52
18. Резултат анализе – Ћирилични текст (језик српски)	53
19. Резултат анализе – Жути билборд (језик српски)	54
20. Резултат анализе – Жути билборд (resize, grayscale)	55
21. Резултат претраге по кључној речи	55
22. Резултат претраге по фрази	56

СПИСАК ТАБЕЛА

<i>Број</i>	<i>Страна</i>
1. Процеси претпроцесирања.....	18
2. Приступи препознавања знакова	19
3. Пример инвертованог индекса	26
4. Поређење развијеног модула са осталим форензичким алатима	30
5. Конфигурација модула	32
6. Покретање OCR-а над сликама	33
7. Прегледање OCR логова	34
8. Претраживање извученог текста.....	35

СПИСАК ЛИСТИНГА

<i>Број</i>	<i>Страна</i>
1. Листинг 1 Основна структура модула.....	39
2. Листинг 2 Препроцесирање помоћу ImageMagick алата	40
3. Листинг 3 Извршење Tesseract OCR и чување у Blackboard	41
4. Листинг 4 Логовање унутар модула.....	43

ЗАХВАЛНИЦА

Захваљујем се ментору Стевану Гостојићу на подршци и смерницама током израде мастер рада.
Хвала и мојој породици и пријатељима на подршци и охрабрењу током целокупних студија.

УВОД

Савремене дигиталне истраге подразумевају анализу великих количина података различитих формата. Један од изазова је идентификација и екстракција текстуалних информација из слика, које нису у машински читљивом облику. У пракси, то отежава претрагу и повезивање релевантних информација, што може довести до споријег и мање ефикасног форензичког процеса. Недостатак модерних уграђених алата за препознавање текста у оквиру Autopsy окружења представља значајан изазов у свакодневном раду форензичара.

У овом раду проблем је решен развојем посебног модула за Autopsy који користи технологију оптичког препознавања карактера (OCR). Као основа примењен је Tesseract OCR, у комбинацији са техникама препроцесирања слике ради побољшања тачности препознавања. Развијени модул омогућава аутоматизовано извлачење текста, његово индексирање и интеграцију са постојећим системом за претрагу (keyword search) и анализу података унутар Autopsy окружења.

Мотивација за овакав приступ произилази из потребе да се унапреди ефикасност дигиталних форензичких истрага. Екстракција текста из слика омогућава бржу идентификацију релевантних информација, смањује ризик од превиђања доказа и проширује могућност анализе. На тај начин, овај рад доприноси не само практичној примени у истражним процесима, већ и даљем развоју алата отвореног кода који се користе у дигиталној форензици.

Структура рада организована је на следећи начин. У другом поглављу описан је софтвер Autopsy и могућност његовог проширења кроз Java и Python модуле. Треће поглавље бави се технологијом оптичког препознавања карактера, индексирањем текста и прегледом других форензичких алата који обављају сличне функције. Четврто поглавље садржи спецификацију функционалних и нефункционалних захтева, као и дизајн развијеног софтвера. Пето поглавље обухвата кључне елементе његове имплементације. Шесто поглавље демонстрира примену модула у пракси, анализира резултате и пружа техничке коментаре о раду софтвера. На крају,

закључно поглавље даје резиме извршених активности, упоређује предности и мане развијеног решења у односу на претходно описане методе и предлаже могућности за даље унапређење

СТАЊЕ У ОБЛАСТИ

Autopsy софтвер - историјат

The Sleuth Kit (TSK) је библиотека и колекција алата командне линије који омогућавају истрагу слика диска (енг. disk images). Главна функционалност TSK библиотеке је могућност анализе система датотека. Библиотека се може уградити у веће алате за дигиталну форензику, а алати командне линије могу се користити за проналажење доказа [1].

Autopsy је један од најраспрострањенијих алата отвореног кода за дигиталну форензику. Првобитно је развијен као графички интерфејс за TSK библиотеку, а данас се користи као самосталан систем за анализу дигиталних доказа. Подржава анализу система датотека, преглед метаподатака, екстракцију артефаката из оперативних система, e-mail клијената и интернет прегледача. Захваљујући архитектури заснованој на модуларности, Autopsy омогућава проширивање функционалности кроз додатке писане у Java и Python програмским језицима [2].

Првобитна идеја је била да се функционалности из командне линије TSK библиотеке учине приступачнијим крајњем кориснику кроз графички интерфејс, олакшавајући процес креирања случаја, управљања извором података и приказивања резултата анализе. Autopsy је у својим првим верзијама већ омогућио да се низ TSK алата покреће над једним случајем и да се резултати прикажу на кориснички уреднији начин [2].

Услед великог броја захтева за анализом великих количина података и развоја нових дигиталних артефаката (веб прегледачи, мобилни формати итд.), Autopsy је временом прерастао у самосталну платформу која укључује индексирање, базу података, извештавање и проширивост кроз додатне модуле. Данашњи Autopsy је изграђен да подржи рад са случајевима (енг. case-centric workflow), индексира текст, генерише временске линије (енг. timeline) и да интегрише различите додатке (енг. plugins) који аутоматизују анализу. Ова еволуција је резултат континуираног развоја, доприноса заједнице и увођења компоненти које решавају специфичне проблеме великих форензичких истрага [2].

Autopsy се користи у различитим окружењима као што су настава и истраживања, мали и велики корпоративни безбедносни тимови и у областима јавног сектора као што су полиција и државне истражне јединице и организације.

Autopsy активно добија нове верзије и допуне. У последњих неколико година видљив је приступ чешћих ажурирања који доносе нове ingest функционалности као што су боље keyword search опције и интегрисано скенирање вируса и малвера. Релативно брз циклус исправки и нових функција указује на то да се пројекат прилагођава новим захтевима у области дигиталне форензике [3].

Autopsy – главне компоненте

Језгро и библиотеке на којима је Autopsy заснован је TSK, који је написан у програмским језицима C/C++ и обавља радње нижег нивоа над системом датотека и сировим медијима. Апликациона логика је написана у Java програмском језику користећи платформу NetBeans, која поверзује TSK слој са слојем корисничког интерфејса (GUI). За индексирање и претрагу Autopsy користи Apache Solr (платформа отвореног кода која садржи функционалности за претраге текста, индексирања и динамичког кластеровања). За сваки случај се може креирати посебна Solr инстанца. За чување и перзистенцију артефаката користи се база података PostgreSQL у вишекорисничким инсталацијама, док се ActiveMQ користи за одређене сервисе. Оваква подела омогућава да се све компоненте развијају потпуно независно, што је важно за дуговечност и проширивост система [4].

Autopsy – инсталација и коришћење

Минимални системски захтеви [5] за инсталацију су:

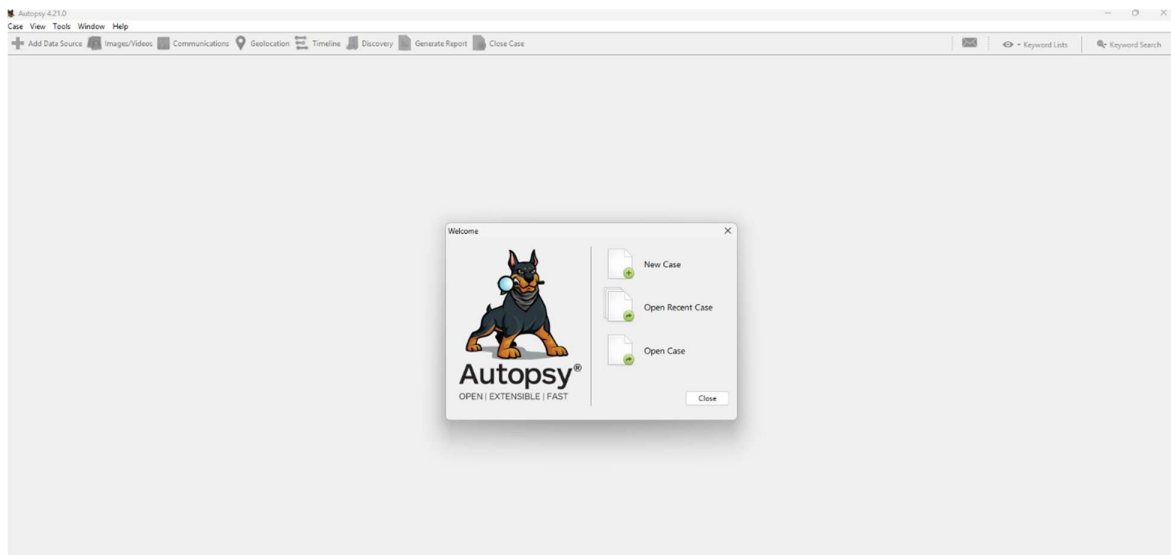
- Оперативни систем: Windows 7/10/11 (64-bit), Linux (Ubuntu, Fedora), macOS (ограничена подршка).
- Хардвер: најмање 4 GB RAM меморије, а препоручује се 8-16 GB за веће скупове података.

- Java окружење: Oracle JDK или OpenJDK, верзија 11 или новија.
- Додатне компоненте: Apache Solr (за индексирање), Python (за скрипте и модуле), NetBeans/Maven (ако се развијају Java додаци).

Званични пакети Autopsy-а доступни су на сајту GitHub репозиторијума. Инсталација се разликује за платформе:

- Windows: Преузима се .exe фајл инсталатор. Инсталатор садржи све потребне зависности укључујући Sleuth Kit и Solr. Након успешне инсталације креира се Autopsy-x.x.x на одабраној путањи (најчешће на C:\Program Files\Autopsy-x.x.x), са унапред дефинисаном структуром фолдера.
- Linux: Преузимање .tar.gz архиве. Распакује се у радни директоријум (нпр. /opt/autopsy). Подешавање JAVA_HOME и додавање bin путање у системски PATH. Инсталација Solr сервиса засебно, као и PostgreSQL.

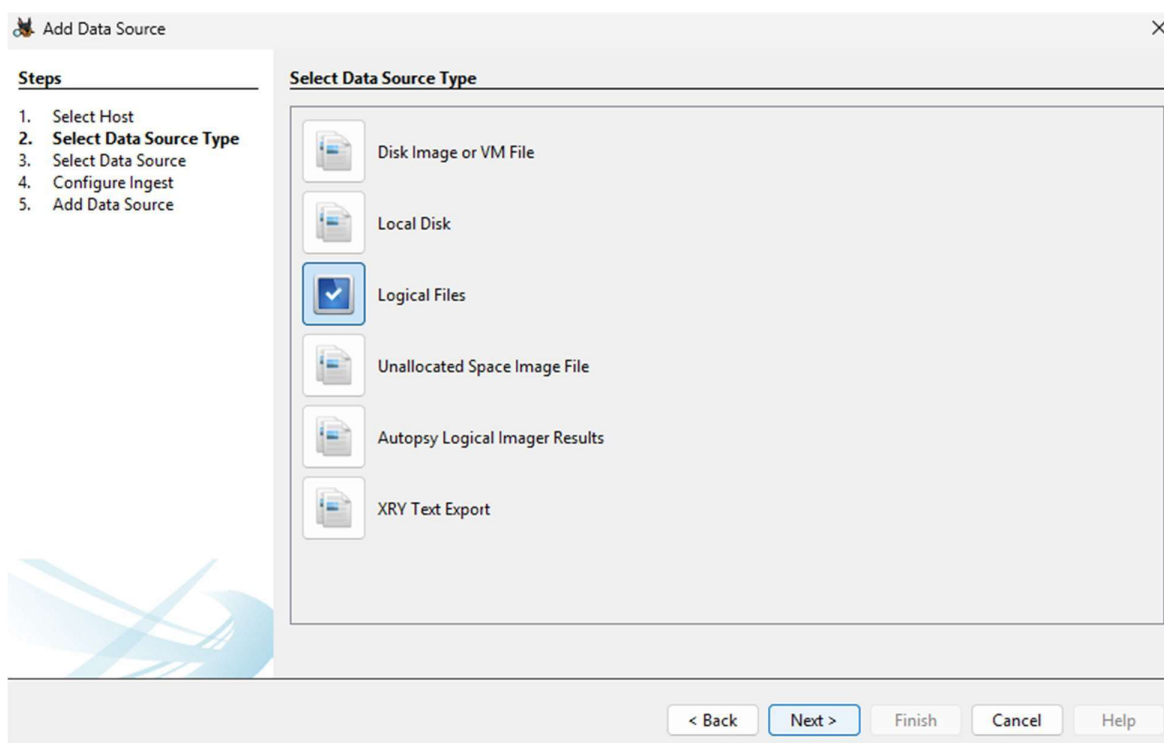
При првом покретању, кориснику се нуди креирање новог случаја (енг. case). На слици 1 приказан је дијалог који се отвара при првом покретању. Омогућава креирање новог случаја или отварање постојећег.



Слика 1. „Чаробњак“ за креирање новог случаја

Отварање већ постојећег случаја се врши кликом на „Open Case“ након чега је потребно одабрати .aut фајл из фолдера претходно креираног случаја. Креирање новог случаја се врши кликом на „New Case“. Потребне информације за креирање новог случаја су путања и име случаја. Опционалне информације су број случаја као и име, презиме, e-mail и белешка истраживача или организације.

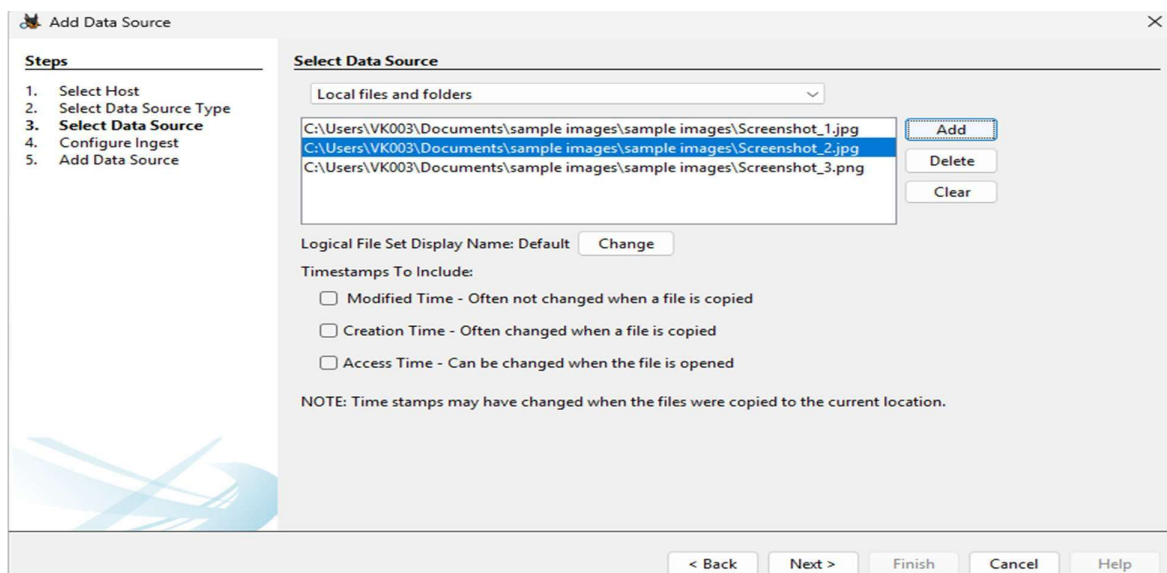
Након успешног креирања новог случаја генерише се нови име домаћина (енг. host name), на основу имена извора података из случаја или се бира већ постојећи. Потом се врши одабир типа извора података. Опције које се овде могу одабрати су приказане на слици 2.



Слика 2. Типови извора података

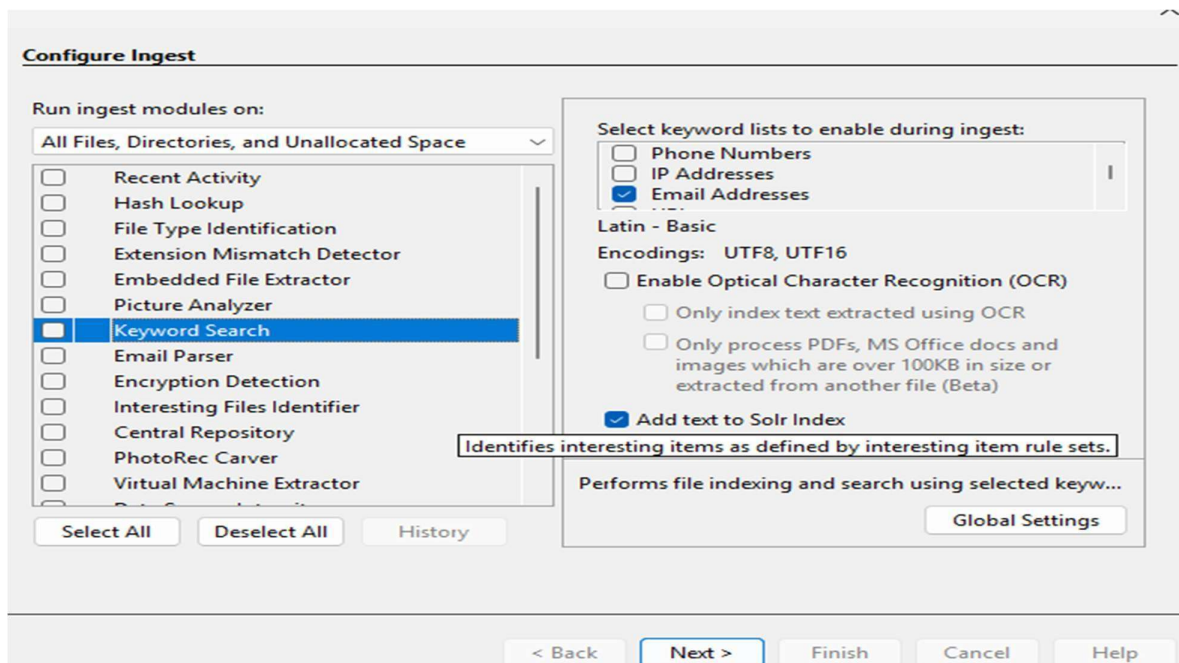
- Disk Image or VM File се користи када истражитељ има целокупну слику диска (енг. disk image) или виртуелне машине. Подржани формати су E01 (EnCase), dd/raw, VMDK (VMware), VHD/VHDX (Hyper-V). Предност је што омогућава потпуну анализу свих партиција диска, система датотека, неалоцираног простора и метаподатака. Најчешћи сценарио је форензичко клонирање хард диска осумњиченог рачунара.

- Local Disk директно приступа физичком или логичком диску који је прикључен на систем. Пример: истражитељ прикључи USB HDD или SSD и дода га као „local disk“. Ова опција је ризичнија јер захтева да се систем чува од измена. Мање се користи у професионалним лабораторијама, али је корисно за брзу анализу.
- Logical Files се користи када су доступни само појединачни фајлови или директоријуми, а не цео диск. Пример: истражитељ добије архиву .zip са e-mail прилозима или скуп логова. Autopsy третира фајлове као податке у оквиру једног случаја и покушава да извуче артефакте. Ово је бржи начин ако нема потребе за анализом целог система датотека. На слици 3 приказан је пример одабира ове опције.
- Unallocated Space Image File, форензичари често издвајају само неалоцирани простор са диска (простор који ОС сматра празним). Овај простор садржи обрисане фајлове и остатке података који се могу вратити. Autopsy омогућава увоз тог дела у виду image фајла. Пример: у истрази брисања докумената, довољно је издвојити само овај сегмент.
- Autopsy Logical Imager Results омогућава избор одређених фајлова или директоријума са диска без прављења целог image-a. Резултат тог процеса је специјалан сет фајлова који се затим може убацити у Autopsy. Предност је мањи обим података, бржа анализа, корисно за ограничене ресурсе или ситуације где није дозвољено копирати цео диск.
- XRY Text Export је комерцијални алат за мобилну форензику (MSAB XRY). Када се мобилни уређај анализира у XRY-у, резултати се могу извести као текстуални извештај. Autopsy може да увезе тај резултат и индексира га као извор података, чиме омогућава претрага и укрштање резултата са другим доказима. Пример: мобилни подаци (SMS, WhatsApp, контакти итд.) анализирани у XRY-у, а затим даље коришћени у Autopsy-ју за повезивање са рачунарским доказима.



Слика 3. Опција Logical files

Када се дода диск, слика (енг. image) или други извор података, Autopsy одмах нуди следећи корак конфигурисање ingest модула. На слици 4 приказан је изглед конфигурације одабира модула.



Слика 4. Одабир Ingest модула

Autopsy – Модули

Autopsy је изграђен као модуларна платформа. Све функционалности се не налазе у основном пакету, већ се додају кроз модуле. Модули омогућавају да се платформа прошири новим алгоритмима, парсерима и специјализованим анализама без измене самог језгра. Сви модули могу се поделити у три велике групе: ingest модули, report модули, general модули [6].

Autopsy – Ingest модули

Ово су најважнији модули јер се извршавају током увоза и анализе података [7]. Они чине такозвани „ingest pipeline“. Постоје две подврсте:

- Data Source Ingest модули, раде над читавим извором података. Пример: Hash Lookup (упоређује све фајлове са познатим hash сетовима), Keyword Search (индексира цео извор у Solr). Углавном се покрећу једном по додавању извора.
- File Ingest модули, раде над појединачним фајловима. Пример: EXIF Parser (извлачи метаподатке из слика). Извршавају се за сваки фајл појединачно у оквиру процеса учитавања.

OCR модул који је тема овог рада је типа File Ingest модул. Анализира сваку слику, врши претпоцесирање и затим извлачи текст и индексира их у бази података.

Autopsy – Report модули

Улога ових модула је генерисање извештаја. Омогућавају извоз резултата у различите формате као што су HTML, CSV, Excel, JSON. Пример: Report Generator модул прави извештај о свим кључним речима пронађеним у случају. Ови модули се покрећу ручно након завршене анализе [6].

Autopsy – General модули

Користе се за специфичне функционалности као што су визуелизација, интеграција са спољним алатима или додатни кориснички интерфејси. Пример: модул који омогућава интерактивно приказивање геолокација на мапи (GIS интеграција) [6].

Предности модуларног система су флексибилност (одабир искључиво релевантних модула), проширивост (лако је написати нови модул) и интеграција (рад са великим бројем спољних алата). Различити тимови могу делити модуле и убрзати развој.

Сви ingest модули, без обзира на врсту, користе Blackboard API да упишу резултате. Blackboard представља централни репозиторијум унутар Autopsy-а где се чувају артефакти као што су кључне речи, hash hit-ови, EXIF подаци итд. Report и General модули могу да користе ове артефакте за даљу обраду, извештаје или визуелизацију.

Autopsy – функционалности

Autopsy је дизајниран као свеобухватна платформа за дигиталне форензичке истраге која омогућава прикупљање, анализу и извештавање података. Његове функционалности покривају широк спектар потреба од анализе система датотека, до индексирања и претраге података. У наставку су описани неки од модула и функционалности.

Анализа система датотека

Autopsy користи TSK као језгро за рад са системима датотека. Подржани су:

- Windows системи: NTFS (New Technology File System), FAT (File Allocation Table)
- Linux системи: EXT (Extended File System)
- MacOS системи: HFS+ и APFS
- Универзални формати: UFS, ISO9660(CD/DVD)

Могућности подразумевају читање целог система датотека из image-а или локалног диска, приказ структуре директоријума и фајлова, приказ метаподатака сваког фајла (датум креирања, датум измене итд.) и идентификација избрисаних фајлова у неалоцираном простору. Ово омогућава корисницима комплетан увид у организацију података на диску, чак и ако је корисник покушао да сакрије или обрише датотеке [1].

Keyword Search модул

Модул за претрагу кључних речи представља један од кључних функционалности унутар Autopsy-а. Омогућава индексирање и претрагу помоћу кључних речи. Користи Apache Solr, систем заснован на Lucene претраживачу, како би створио структуре података погодне за претрагу. Основни концепт у овом процесу је инвертовани индекс.

Уместо да се памти сваки документ са својим садржајем, гради се структура која за сваку реч чува листу докумената у којима се та реч појављује. Захваљујући оваквом индексу, претрага по кључним речима постаје изузетно брза [8].

Употребом претраге по кључним речима, корисник одмах добија све релевантне резултате за унет критеријум претраге. Подржани су разни формати фајлова које је могуће претраживати (нпр. PDF, docx итд.).

EXIF метаподаци и анализа мултимедијалних датотека

Један од значајнијих извора доказа у дигиталној форензици су мултимедијални фајлови, пре свега фотографије и видео снимци. Ови фајлови не садрже само визуелни садржај, већ у себи могу носити и велики број метаподатака који откривају околности под којима је датотека настала [9].

Најпознатији стандард за чување таквих података је EXIF (Exchangable Image File Format). Метаподаци се аутоматски уписују у фотографију или видео у тренутку снимања, најчешће од стране паметних телефона или дигиталних камера.

Ови подаци могу укључивати временску ознаку (тачан датум и време када је фотографија снимљена), подаци о камери (модел уређаја, произвођач итд.), параметри снимања (експозиција, брзина затварача, баланс беле боје), геолокација (GPS координате), thumbnail (умањена верзија слике која се често користи за бржи приказ, корисно када је оригинални садржај оштећен).

Поред EXIF-a, Autopsy може да изврши и класификацију мултимедијалних датотека. Постоје и могућности аутоматског препознавања недозвољених садржаја коришћењем hash листа познатих датотека (нпр. база познатих фотографија незаконитог садржаја).

E-mail анализа

E-mail садржаји представљају један од најчешће коришћених извора података у дигиталним истрагама јер садрже комуникацију, прилоге, временске ознаке и метаподатке који помажу у реконструкцији догађаја. У форензичком контексту, анализа ових врста садржаја обухвата посматрање садржаја порука, заглавља (headers), прилога репозиторијум адресара, метаподатака и логова mail сервера [10].

Технички формат подразумева стандарде као што су RFC 822/2822 за структуру заглавља и MIME (Multipurpose Internet Mail Extensions) за кодирање мултимедијалних прилога и сложених тела порука. У пракси се јављају формати као што су .eml фајлови, .mbox архиве, .ost или .pst (комерцијални формати које користи Microsoft Outlook-a). Форензички алати морају подржати све ове формате јер је структура и начин чувања мејлова често критичан за коректност анализе.

Autopsy пружа парсере за најчешће формате порука и омогућава да се извучени садржај и метаподаци унесу у систем као артефакти који могу бити индексирани и повезани са другим елементима случаја.

Идентификација типа фајла (File Type Identification)

У информатици, фајл има две компоненте које се користе за његову идентификацију: екстензију и унутрашњу структуру. Форензичка анализа подразумева анализу унутрашње структуре јер екстензије могу бити погрешне или намерно промењене.

Најчешћи начин аутоматске идентификације базира се на file signatures (унапред дефинисани бајт шаблони који се налазе на почетку фајла).

Autopsy користи комбинацију TSK-ових механизма да би извршио идентификацију типа фајла током ingest фазе. Ово омогућава да се ресурси као што су време и CPU троше смислено и да се резултати форензичке анализе заснивају на правилно претпостављеној природи садржаја [11].

Проширивост Autopsy окружења са Python и Java модулима

Једна од најважнијих предности Autopsy алата у контексту дигиталне форензике је његова висока проширивост. Подржава проширења кроз модуле који се могу развијати у Python или Java програмским језицима. Java модули интегришу нове форензичке анализе директно у платформу, док Python модули омогућавају експерименталну или скриптовану обраду података, што је идеално за тестирање.

Java модули

Java модули се додају одабиром опције Plugins унутар Autopsy-а. Тамо је потребно одабрати и инсталирати .nbm фајл модула који је претходно развијен [12].

Java модули у окружењу обично реализују сложеније функционалности, као што су:

- Индексирање и претрага специфичних формата датотека,
- Дубинска анализа докумената, слика и мултимедијалних садржаја,
- Интеграција са базама података и другим комерцијалним форензичким алатима.

Иако Java модули у Autopsy-у омогућавају интеграцију сложених и стабилних форензичких функционалности, постоје одређене мане и изазови који могу утицати на развој и примену ових модула.

- Комплексност развоја. Java модуле није лако креирати, нарочито за кориснике који нису упознати са Java програмским језиком или са унутрашњом архитектуром Autopsy платформе.

- Проблеми са компатибилношћу. Autopsy користи специфичну верзију Java окружења (JDK11 или JDK17). Модул који је развијен за једну верзију може бити некомпатибилан за друге верзије што захтева додатно тестирање и адаптацију.
- Зависност од великог броја спољашњих JAR датотека који често због некомпатибилности верзија доводе до *ClassNotFoundException* или *NoClassDefFoundError* грешака током покретања модула. Све зависности морају правилно бити додате у class path водећи рачуна о међусобној компатибилности верзија JAR датотека које се додају.

Python модули

Python модули пружају већу флексибилност и брзину развоја софтвера, што их чини идеалним за истраживаче и академске пројекте. Python модул је лако написати за аутоматизацију одређених задатака као што су претрага, филтрирање, претварање или нормализација, извлачење метаподатака и интеграција са другим аналитичким алатима [13].

Autopsy користи тзв. „Jython“ имплементацију Python програмског језика на Java виртуелној машини. То значи да Python код није изворно извршен од стране класичног Python интерпретера, већ се преводи у Java бајт-код и покреће унутар Java окружења Autopsy-a.

Предности развијања ове врсте модула су брзина развијања и једноставност интеграције у окружење. За Python модул је потребан било који текст едитор у ком ће се писати код. Након успешно завршене имплементације потребно је отворити фолдер из Autopsy окружења бирајући опцију Tools > Python Plugins.

Окружење ће отворити фолдер у којем је потребно креирати нови фолдер са одговарајућим именом модула који се развија. Унутар креираног фолдера је потребно креирати или копирати Python скрипту где је имплементиран модул.

Мане развијања ове врсте модула су ограничена подршка за Python библиотеке, спорија брзина извршавања и потреба за познавањем Autopsy Java API-a. Најновија верзија Autopsy окружења подржава Python верзију 2.7.18. Верзије као што су Python 3 и новије верзије нису компатибилне са окружењем иако је компанија TSK најавила да активно раде на интеграцији са новијим верзијама Python-a.

Пошто Jython ради преко Java виртуелне машине, за интензивне операције може бити спорији од класичног интерпретера. Да би модул користио Autopsy функционалности као што су Blackboard и TSK библиотеке, програмер мора да разуме како Jython користи Java класе писане за Autopsy.

Поређење са сличним алатима

У области дигиталне форензике постоји више алата који се користе за анализу података са компјутера, мобилних уређаја и других дигиталних медија. Сваки од ових алата има своје предности и мане, а избор зависи од потреба корисника, буџета и типа истраге. У наставку се врши кратак упоредни преглед три алата: EnCase, FTK, X-Ways Forensics са Autopsy-ем.

EnCase је комерцијалан алат који представља индустријски стандард у дигиталној форензици. Он омогућава дубинску анализу података, укључујући извлачење и индексирање датотека, анализу е-маилова, као и подршку за оптичко препознавање карактера из слика и докумената. Највећа предност је његова широко прихваћена употреба у судским процесима. Међутим, EnCase је скуп, што ограничава његову доступност малим лабораторијама или академским институцијама [14].

FTK (Forensic Toolkit) је још један комерцијални алат, познат по снажним могућностима за претрагу кључних речи и анализу дигиталних доказа са компјутера, мобилних уређаја и cloud извора. Издваја се по својој брзини индексирања и кориснички пријатељском интерфејсу, што омогућава ефикасно претраживање великих скупова података. Један од недостатака је висока цена лиценци и потреба за релативно моћним хардвером за оптималне перформансе [15].

X- Ways Forensics је алат који се често описује као брз и лаган, са ниским системским захтевима. Иако је веома ефикасан у дубинској анализи и може да изврши напредне форензичке операције, он је мање „user friendly“ у поређењу са EnCase и FTK. Интерфејс може бити компликован за

нове кориснике, а документација је понекад недовољно објашњена. Међутим, за искусне форензичаре представља веома флексибилан и ефикасан алат [16].

ОПТИЧКО ПРЕПОЗНАВАЊЕ КАРАКТЕРА И ИНДЕКСИРАЊЕ ТЕКСТА

OCR кроз историју

Оптичко препознавање знакова (енг. Optical Character Recognition – OCR) представља технологију која омогућава аутоматско претварање текста са слика или скенираних докумената у машински читљив и обрадив формат. Основна идеја OCR-а јесте да се визуелни приказ знакова, који човек може да прочита, преведе у дигиталну репрезентацију погодну за даљу обраду, претраживање и чување у базама података. Овај процес је од кључне важности у контексту дигитализације докумената, јер омогућава елиминацију ручног прекуцавања и значајно убрзава приступ великој количини информација [17].

Историјски гледано, OCR системи су започели свој развој средином 20. века. Први покушају били су засновани на шаблонском препознавању, где је сваки знак морао бити упоређен са унапред дефинисаним обрасцем. Иако примитиван, овај приступ је поставио темеље за даљи развој. Током 1980-их и 1990-их година, развој статистичких метода и алгоритама машинског учења омогућио је већу флексибилност и прецизност. Савремени системи користе дубоке неуронске мреже, конволуционе архитектуре и методе секвенцијалног моделирања, што им омогућава препознавање текста чак и у условима лошег квалитета, рукописа или мултијезичких докумената [18].

Значај OCR-а посебно се огледа у његовој широкој примени. У библиотекама и архивама ова технологија омогућава дигитализацију и индексирање књига, новина и историјских докумената. У административним и пословним системима користи се за аутоматску обраду рачуна, фактура, формулара и службене документације. У домену саобраћаја примена OCR-а уочава се код система за препознавање регистарских таблица (ANPR), док у здравству омогућава дигитализацију медицинске документације. Посебно је значајна улога OCR-а у дигиталној форензици, где служи за извлачење текста из слика, PDF датотека или скенираних доказа, како би се добијене информације касније индексилале и претраживале током истрага.

Иако је OCR данас изузетно напредовао, остају бројни изазови. Слаба резолуција, искошени документи, рукопис или текстови на језицима са сложеним писмима и даље представљају препреку. Међутим, трендови показују да интеграција OCR-а са модерним техникама вештачке

интелигенције отвара нове могућности за високо прецизну, брзу и скалабилну обраду текста из визуелних извора.

Основни кораци у OCR процесу

Процес оптичког препознавања знакова може се поделити у неколико кључних фаза. Свака од њих има за циљ да постепено трансформише улазну слику документа у машински читљив текст са што мање грешака. У наставку ће бити описане четири основне фазе: претпроцесирање, сегментација, препознавање и постпроцесирање [17].

Претпроцесирање слике

Први корак у OCR-у је припрема слике тако да она буде погодна за даље анализе. Сирове слике често садрже шум, искошење, неравномерно осветљење или ниску резолуцију, што значајно утиче на тачност препознавања. У табели 1 описани су основни процеси претпроцесирања [18].

Процес	Опис
Бинаризација	Процес претварања слике у црно-белу. Циљ је да текст постане одвојен од позадине.
Филтрирање шума	Уклањање мрља, прашина или дигиталних артефаката.
Исправљање искошења (deskew)	Уколико је документ скениран под углом, неопходно је исправљање ради правилне сегментације редова.
Нормализација резолуције	OCR системи оптимално раде са резолуцијом од 300 dpi (dots per inch), па се често примењује.
Побољшање контраста	Повећање разлике између слова и позадине.

Табела 1. Процеси претпроцесирања.

Сегментација

Сегментација подразумева раздвајање текста на мање целине, редове, речи и појединачне знакове. Она представља један од најизазовнијих корака.

За детекцију текста користе се алгоритми који анализирају хоризонталну пројекцију пиксела како би идентификовали појединачне редове. Након тога, раздвајање речи се најчешће врши анализом размака између пиксела унутар сваког реда. Сегментација карактера је процес идентификације граница између појединачних слова. Ово је критично код језика са спојеним писмом (нпр. арапски), као и код рукописа.

Неуспешна сегментација доводи до тога да OCR систем препозна више слова као једно или обрнуто, што утиче на коначну тачност.

Препознавање знакова

Историјски гледано, развијена су три главна приступа препознавања знакова. У табели 2 су приказани и појашњени кључни приступи.

Приступ	Опис
Шаблонско препознавање (template matching)	Најстарија техника, где се слика знака упоређује са унапред дефинисаним шаблонима. Ограничење је у томе што варијације у фонтовима или величинама доводе до грешака.

Статички модели	Употреба техника као што су скривени Марковљеви модели (HMM) или Support Vector Machines (SVM), које омогућавају већу робусност.
Неуронске мреже и deep learning	Савремени OCR системи попут Tesseract-а користе конволуционе неуронске мреже (CNN) и LSTM мреже које уче зависности у секвенцама. Ово омогућава препознавање чак и у условима лошег квалитета или код сложених писама.

Табела 2. Приступи препознавања знакова

Прецизност модерних deep learning система често прелази 95%, што их чини применљивим у већини реалних сценарија [19].

Постпроцесирање

Постпроцесирање има за циљ исправљање грешака и унапређење квалитета добијеног текста. Коришћење речника како би се извршило упоређивање препознате речи са базом података познатих речи и аутоматски исправља очигледне грешке. Контекстуална анализа (n-grams и статистичке анализе) користи се да би се реч уклопила у контекст. Комбиновање више OCR система у неким случајевима даје боље резултате, јер различити системи греше на различитим местима.

Све наведене фазе заједно чине један логички ланац који омогућава да се визуелни садржај претвори у дигитални текст. Уколико једна од фаза не функционише како треба, коначни резултат ће бити значајно мање употребљив. Зато је OCR често посматран као интеграција више дисциплина као што су обрада слике, машинско учење и обраде природног језика.

Савремене технологије у OCR-у

Савремени системи за оптичко препознавање карактера значајно су напредовали последњих деценија и данас представљају један од кључних елемената у процесима дигитализације докумената, обраде текста и аутоматизације административних послова. Постоји велики број софтверских решења која се могу сврстати у две главне групе, алати отвореног кода и комерцијални алати. Свако од ових решења има своје предности, мане и специфичне области примене. У наставку су представљене најважније OCR технологије које се данас користе у пракси.

Tesseract OCR

Tesseract је један од најпознатијих OCR алата отвореног кода, првобитно развијен од стране Hewlett Packard-а осамдесетих година, а касније унапређен и одржаван од стране Google-а. Овај систем подржава велики број језика, укључујући и ћирилицу, што га чини погодним за примену у различитим културним и језичким контекстима. Једна од највећих предности Tesseract-а је могућност тренирања сопствених модела, односно прилагођавање систему специфичним фонтовима или рукописима. Иако је у прошлости био ограничен по питању тачности, интеграција са deep learning библиотекама донела је значајно побољшање у прецизности препознавања знакова и карактера.

ABBYY FineReader

ABBYY FineReader је комерцијални OCR алат који се сматра једним од најпрецизнијих у својој категорији. Његова највећа предност није само у тачности препознавања карактера, већ и у способности очувања структуре докумената. FineReader успешно препознаје табеле, хијерархију заглавља, више колона и чак уграђене слике, чиме омогућава да дигитализовани документ остане прецизна реплика оригинала. Ово је од изузетног значаја у правним и административним процесима где је битно не само шта се пише у документу већ и како је документ форматиран [21].

Cloud OCR решења

Последњих година дошло је до појаве бројних cloud OCR услуга које нуде додатне могућности у односу на класичне standalone апликације. Најпознатији представници су Google Vision API, Microsoft Azure OCR и Amazon Textract. Ове услуге се заснивају на инфраструктури великих cloud провајдера, што им омогућава високу скалабилност, константно унапређивање модела и интеграцију са другим AI сервисима као што су превођење, класификација или препознавање објеката на сликама. Поред тога, cloud решења елиминишу потребу за локалним хардверским ресурсима, што их чини погодним за организације које желе брзо и флексибилно увођење OCR-а без значајних почетних инвестиција [22] [23].

Handwriting OCR (ICR – Intelligent Character Recognition)

Један од највећих изазова у области OCR-а је препознавање рукописа, познато и као Intelligent Character Recognition (ICR). За разлику од штампаног текста, рукописи се карактеришу великом варијабилношћу у облику слова, неуједначеном величином и нагибом, као и честим спајањем карактера. Иако традиционални OCR системи често показују слаб резултат у овој области, савремени deep learning приступи, а нарочито конволуционе и рекурентне неуронске мреже, показују значајан напредак. Ипак, прецизност и даље варира у зависности од квалитета улазног материјала, стила писања и језичког контекста [24].

У целини, може се закључити да савремене технологије OCR-а нуде широк спектар решења, од бесплатних и флексибилних алата, преко високо-прецизних комерцијалних алата, до cloud услуга које интегришу OCR са другим AI могућностима. Посебно изазовна, али и перспективна област је препознавање рукописа, која се све више ослања на deep learning технике. Оваква разноврсност међу алатима омогућава корисницима да изаберу технологије које најбоље одговарају њиховим потребама.

Примене OCR-а

Оптичко препознавање карактера данас има изузетно широк спектар примена које се простиру од свакодневних административних послова до специјализованих форензичких анализа.

Захваљујући напретку у прецизности и брзини препознавања, OCR је постао један од кључних алата у дигитализацији докумената и трансформацији аналогних садржаја у дигитални облик погодан за претрагу и анализу. У наставку су представљене најважније области примене OCR технологије.

Дигитализација књига и архива

Једна од првих и најраспрострањенијих примена OCR-а јесте дигитализација библиотека и архива. OCR омогућава да се скениране књиге, новински чланци или историјски документи претворе у дигитални текст који се може претраживати, индексирати и архивирати. На овај начин се значајно олакшава приступ културном и научном наслеђу, као и његова заштита од физичког пропадања. Посебан изазов у овој области представљају стари рукописи и оштећени документи, где је неопходна примена специјализованих OCR модела.

Административни системи

OCR има важну улогу у аутоматизацији административних процеса, посебно у областима као што су рачуноводство и обрада личних докумената. На пример, системи за управљање фактурама користе OCR како би аутоматски извукли релевантне податке (број фактуре, датум, износ, ПИБ компаније) и унели их у финансијске системе. Слично томе, у државној администрацији OCR се примењује за обраду скенираних личних докумената, образаца и формулара, чиме се смањује потреба за ручним уносом података и смањује ризик од грешака.

Препознавање регистрационих таблица (Automatic Number Plate Recognition)

Један од основних елемената за аутоматско препознавање регистарских таблица на возилима је OCR. Ове системе користе саобраћајна полиција, системи наплате путарина, као и у контролама приступа на паркинзима и безбедносно осетљивим зонама. Препознавање таблица захтева високу прецизност и временске прилике. Захваљујући интеграцији OCR-а са видео аналитиком и системима машинског учења, модерни ANPR системи достижу високу прецизност у практичним сценаријима.

Банкарство

У банкарском сектору OCR је постао стандард за обраду документа као што су чекови, уговори или кредитна документација. Аутоматска обрада чекова подразумева препознавање бројева рачуна, износа и потписа, што значајно убрзава процес трансакција и смањује ризик од људских грешака. Слично томе, OCR се користи у дигитализацији уговорне документације, што омогућава брзо претраживање и прављење база знања у банкама и финансијским институцијама.

Медицинска документација

OCR омогућава дигитализацију медицинских картона, лабораторијских налаза и извештаја. На овај начин медицинске установе могу ефикасније чувати, претраживати и делити информације о пацијентима. Посебан изазов у овој области представља препознавање рукописа лекара и белешки доктора, које често садрже специфичне медицинске термине и скраћенице.

Форензичке примене

У дигиталној форензици, OCR се користи за анализу скенираних докумената, рукописа и доказа добијених са фотографија, видео записа или скенера. У оквиру истрага, OCR омогућава да се из великих количина слика извуче текст који се касније може индексирати и претраживати по кључним речима. Ово значајно убрзава процес анализе, омогућава идентификацију релевантних информација и повећава вероватноћу откривања кључних доказа. OCR резултати

могу бити прихваћени као доказни материјал на суду, под условом да је технологија адекватно верификована и да је поступак прикупљања података извршен у складу са форензичким стандардима и правилима.

OCR је постала универзална технологија која налази примену у готово свим индустријама и институцијама. Њена улога није ограничена само на дигитализацију, већ све више постаје интегрални део паметних система за обраду информација и подршку одлучивању.

Индексирање текста након OCR-а

Када се текст извуче из слика или PDF докумената путем OCR-а, он постаје доступан у електронском облику. Међутим, такав „сирови текст“ није довољно практичан за ефикасну употребу. Уколико је потребно сваки пут прелиставати цео текст приликом претраге кључних речи, време обраде постаје неприхватљиво велико, посебно у случајевима где се ради о великим скуповима докумената. Зато се користи индексирање текста, поступак којим се текст организује у структуру која омогућава брзо и ефикасно претраживање [25].

Индексирање представља процес креирања помоћних структура података које омогућавају да се одређена реч или фраза пронађе унутар великог скупа докумената у врло кратком временском року. Представља основу свих савремених search engine технологија. У дигиталној форензици, индексирање препознатог текста омогућава брзу идентификацију релевантних доказа у било ком формату.

Инвертовани индекс представља основни концепт најчешће коришћеног алгоритма. За разлику од класичног индекса у књизи, који за сваку тему показује на којим се странама налази, инвертовани индекс гради мапу у којој је кључ реч, а вредност је листа докумената (и позиција унутар докумената) у којима се та реч појављује. На овај начин је могуће одмах приступити релевантним документима који садрже одређени појам, без потребе за линеарним тражењем и прегледом целог скупа докумената или случаја.

Индексирање текста након OCR-а обично укључује неколико стандардних корака:

1. Токенизација – раздвајање текста на појединачне речи или токене.
2. Нормализација – претварање свих речи у мали регистар и уклањање дијакритика (нпр. Џ у С).
3. Stemming и lemmatization – свођење речи на њихов корен или основни облик (нпр. „пишем“, „писао“, „писање“ у „пис“).
4. Уклањање стоп речи – елиминација честих и мање значајних речи („и“, „или“, „да“, „је“).
5. Генерисање мапе – креирање инвертованог индекса у форми: реч – [документ1, позиција1], [документ2, позиција2], ...

У табели 3 приказан је пример инвертованог индекса.

Реч	Идентификатор документа и позиције
уговор	Документ1(15), Документ2(87), Документ5(42)
банка	Документ2(13), Документ3(5), Документ7(98)
рачун	Документ1(67)

Табела 3. Пример инвертованог индекса

Постоји више алата и библиотека које омогућавају индексирање текста:

- Lucene – један од најпознатијих система отвореног кода за индексирање и претрагу.
- Apache Solr – надоградња на Lucene, која нуди веб интерфејс и напредне функције.
- Elasticsearch – дистрибуирани систем за индексирање и претрагу, базиран на Lucene-у, који се често користи у Big Data окружењима.
- Autopsy – уграђује Lucene као свој основни индексирајући механизам, чиме омогућава претрагу по тексту који је претходно извучен из датотека.

Овим поступком, огромна количина неструктурираних података добијеног из OCR-а постаје организована и лако претражива. То чини индексирање неопходним кораком за сваку примену OCR-а, било да је реч о претраживању дигиталних архива или форензичкој анализи потенцијалних доказа.

Проблеми индексирања после OCR процеса представљају велик изазов. Такозвани „OCR noise“ настаје услед лошег квалитета скенираних докумената, искривљености текста, фонтова који нису добро подржани или због присуства рукописа. Као резултат тога, уместо исправне речи „уговор“ систем може препознати „угѠвор“. Овакве грешке отежавају класично индексирање јер се такви термини неће појавити у индексу у исправном облику. Да би се ово ублажило, користе се технике fuzzy претраге, које омогућавају проналажење појмова са одређеним бројем грешака или разлика у словима. Још један проблем су језичке специфичности. У српском језику исти појам може имати више облика због падежа (нпр. „банка“, „банке“, „банци“), што захтева примену лематизације или морфолошке анализе. Слично важи и за дијакритике (č/ć/š/ž/đ), јер без нормализације различити облици исте речи могу бити погрешно третирано као различити појмови. Ови изазови значајно утичу на тачност индексирања и захтевају прилагођене алгоритме који могу да обраде несавршености текста.

Да би се превазишла ограничења класичног индексирања, развијене су напредне технике које унапређују прецизност и ефикасност претраге. Једна од њих је n-gram индексирање, које разлаже речи на мање делове (n-grams), омогућавајући проналажење и делимичних поклапања. Овај приступ је посебно користан код OCR текста са грешкама, јер чак и ако је један карактер погрешно препознат, постоји велика вероватноћа да ће се остали делови речи поклопити са траженим појмом. Други приступ је индексирање по фазама, где се уместо појединачних речи у индекс уносе и целе фразе као посебне јединице. То је важно јер појединачна реч често нема довољно контекста (нпр. „рачун“ може бити банкарски или телефонски), док фраза „банкарски рачун“ јасно указује на специфичан домен. Системи за индексирање се све више комбинују са техникама обраде природног језика (NLP) и машинског учења. Хибридни модели користе класичан инвертовани индекс за прецизно претраживање, али истовремено примењују и векторске репрезентације речи или трансформер моделе како би омогућили семантичку претрагу. На овај начин, систем може пронаћи документ који садржи реч „банка“ чак и када је корисник претраживао појам „финансијска институција“, јер постоји семантичка блискост.

Како количина документа расте, индекси могу постати велики и заузимати значајан простор у меморији или на диску. Због тога се примењују различите технике оптимизације индекса. Једна од најчешће коришћених метода је компресија индекса, којом се листе докумената у инвертованом индексу складиште у сажетом облику, што смањује потребу за простором, али убрзава и читање јер се више података може држати у кеш меморији. Популарна техника за овакав приступ је Huffman кодирање, које омогућава ефикасно складиштење великих листа идентификатора документа. Други важан аспект је управљање великом количином података. У окружењима где се ради са милионима или милијардама докумената, класично индексирање није довољно. У тим случајевима користе се дистрибуирани системи попут Elasticsearch, који омогућавају да се индекс подели на више чворова у кластеру. Ово не само да повећава брзину претраге, већ обезбеђује и високу доступност система у случају отказа појединих сервера. Поред тога, индекси се могу оптимизовати и у складу са конкретним потребама нпр. у форензичкој анализи се често креирају специјализовани индекси који садрже само одређене типове података (мејлови, бројеви телефона, IP адресе итд.), чиме се додатно убрзава претрага у истражном процесу.

OCR и индексирање у форензичким алатима

Модерни форензички софтвери препознају да велики део доказа није у класичном текстуалном облику, већ унутар слика, PDF докумената и скенираних прилога. Управо због тога су у своје оквире интегрисали модуле за оптичко препознавање карактера, као и механизме за индексирање текста, чиме истражитељима и корисницима омогућавају брзо претраживање и анализу.

EnCase Forensic један од најраспрострањенијих комерцијалних форензичких алата, подржава OCR као додатак у својим верзијама намењен за корпоративне и истражне сврхе. OCR функционалности у EnCase-у омогућава да се извуку текстуални подаци из слика и PDF докумената, који се потом индексирају унутар EnCase базе података. Индексирани садржај истражитељу омогућава претрагу по кључним речима, што је кључно у великим истрагама где је потребно брзо пронаћи специфичне појмове. Овај приступ је веома сличан теми овог рада, где и развијени OCR модул за Autopsy издваја текст са слика уз помоћ Tesseract OCR и чува унутар интерне базе података, одакле је текст доступан за претрагу.

FTK (Forensic Toolkit) integriше OCR још у ранијим верзијама, при чему користи ABBYY FineReader као мотор за препознавање текста. Ово га чини веома поузданим, јер ABBYY FineReader представља један од најпрецизнијих комерцијалних OCR система. Када се OCR примени на документ, резултати се чувају унутар FTK базе и истражитељ их може претраживати заједно са осталим доказима. FTK такође креира индексе, што значи да није потребно сваки пут пролазити кроз цео текст, већ је претрага практично тренутна. У табели 4 приказано је поређење развијеног модула за Autopsy са осталим постојећим форензичким алатима.

Алат/модул	Тип алата	Подржане функције	OCR подршка	Индексирање и претрага	Предности	Ограничења
Развијен OCR модул за Autopsy	Python модул додатак за Autopsy	OCR над сликама Препроцесирање са ImageMagick Подршка више језика Складиштење резултата у Blackboard ради претраге	Tesseract интеграција	Резултати се индексирају у Autopsy Blackboard	Потпуна интеграција у Autopsy Флексибилно подешавање	Захтева ручну инсталацију ImageMagick и Tesseract
Autopsy Keyword Search Ingest Module	Уграђени модул у Autopsy	Претрага целог текста Индексирање садржаја датотека	Не	Да, Apache Solr индекс	Веома брзо претраживање великих сетова података	Не ради OCR већ само тражи већ постојећи текст
X-Ways Forensics	Комерцијални алат	Анализа докумената	Да	Да	Ефикасан и једноставан	Комерцијалан, ограничена подршка у

		OCR преко додатних компоненти		Сопствени индексни механизам		односу на Tesseract
FTK	Комерцијани форензички пакет	Напредно индексирање OCR преко FTK Imager опције	Да, интегрисан	Централни индекс за брзу претрагу	Веома добро решење за корпорације	Захтева снажне ресурсе и лиценцу
EnCase	Комерцијалан алат	Претрага по целом тексту Keyword highlight OCR (плаћени модули)	Да, преко add-on модула	Да	Веома поуздан и признат у судској пракси	Скупо и затворено решење
PDF OCR Tools	Алат специјално за OCR	OCR над PDF документима	Да	Не	Једноставан, брз	Ради само са PDF документима

Табела 4. Поређење развијеног модула са осталим форензичким алатима

Из табеларног приказа 4 може се закључити да је развијени Python модул отвореног кода прилагођен за Autopsy окружење, флексибилан, али захтева више ручног подешавања у односу на комерцијалне алате.

СПЕЦИФИКАЦИЈА

У овом поглављу је представљена спецификација развијеног софтверског модула, која обухвата дефинисање захтева, дизајн и кључне елементе имплементације. Спецификација има за циљ да јасно прикаже шта систем треба да ради, како је структурисан и на који начин описује интеракцију корисника са системом, као и дијаграми који формално представљају архитектуру и токове података. На овај начин омогућава се боље разумевање улоге модула у ширем контексту дигиталне форензике.

Фокус овог поглавља биће на спецификацији захтева кроз дијаграме случајева коришћења и описе појединачних случајева, као и на спецификацији дизајна путем дијаграма класа, распореда и секвенци. Сврха је да се формално представи структура система, односи између компоненти и токови интеракције између корисника и софтвера. Оваква спецификација омогућава разумевање тренутне верзије модула и основу за његово даље одржавање и надоградњу.

Поглавље такође описује кључне елементе имплементације, укључујући коришћене платформе, радне оквири и софтверске алате. Овде се наводе и битни фрагменти изворног кода, који илуструју начин на који софтвер обрађује слике, покреће OCR, обрађује резултате и креира артефакте у Autopsy Blackboard-у.

Дијаграм случајева коришћења

Дијаграми случајева коришћења или „use-case“ дијаграми представљају визуелну репрезентацију интеракције корисника са системом. Они омогућавају да се на прегледан начин прикажу све функције које модул пружа, као и односи између различитих активности корисника. За OCR модул у Autopsy-у, дијаграми случајева коришћења илуструју како форензички аналитичар покреће обраду слика, добија текстуалне резултате, снима их у Blackboard и врши претрагу и преглед резултата. У наставку су приказани поменут дијаграм и његови описи.



Слика 5. Дијаграм случајева коришћења

Дијаграм са слике 5 обрађен је у наредним табелама.

Назив случаја коришћења	Конфигуриши модул
Иницијатор	Дигитални форензички аналитичар
Циљ	Постављање параметара OCR модула пре обраде фајлова (подржани формати, језик, опције претпроцесирања).
Предуслови	OCR модул је доступан у систему.
Опис	Корисник одабира које типове фајлова ће обрадити, који језик ће се користити и да ли ће се применити претпроцесирање слика (grayscale или resize).
Постуслови	Подешавања су сачувана и модул је спреман за покретање OCR-а.
Основни ток	<ol style="list-style-type: none"> 1. Корисник отвори поставке модула 2. Корисник одабере типове фајлова 3. Корисник одабере језик OCR-а 4. Корисник одабере опције претпроцесирања 5. Сачува подешавања

Алтернативни ток	Корисник може изабрати подразумеване вредности без измена.
-------------------------	--

Табела 5. Конфигурација модула

Назив случаја коришћења	Покрени OCR над сликама
Иницијатор	Дигитални форензички аналитичар
Циљ	Извлачење текста из подржаних фајлова и складиштење резултата у Autopsy Blackboard
Предуслови	OCR модул је конфигурисан и слике/фајлови су доступни.
Опис	Корисник покреће OCR на изабраним фајловима. Сваки фајл пролази кроз претпроцесирање, а резултат се снима у Blackboard за даљу анализу.
Постуслови	Текст је извучен и сачуван у систему. Логови обраде су доступни.
Основни ток	<ol style="list-style-type: none"> 1. Корисник изабере фајлове 2. Модул претпроцесира слике са ImageMagick-ом 3. Покреће се Tesseract OCR. 4. Резултати се снимају у Blackboard. 5. Логови се генеришу.
Алтернативни ток	Ако OCR није успео за одређени фајл, генерише се порука о грешци у логовима.

Табела 6. Покретање OCR-а над сликама

Назив случаја коришћења	Прегледа OCR логове
Иницијатор	Дигитални форензички аналитичар
Циљ	Праћење статуса обраде и грешака приликом OCR-а.
Предуслови	OCR је покренут и логови су креирани.
Опис	Корисник прегледа логове да би видео који фајлови су успешно обрађени, а који имају проблеме.
Постуслови	Корисник има увид у успешност и грешке OCR процеса.
Основни ток	<ol style="list-style-type: none"> 1. Корисник отвори логове 2. Прегледа листу обрађених фајлова 3. Прегледа поруке о грешкама
Алтернативни ток	Могуће филтрирање логова по фајлу или статусу обраде.

Табела 7. Прегледање OCR логова

Назив случаја коришћења	Претражи извучени текст
Иницијатор	Дигитални форензички аналитичар
Циљ	Брзо пронаћи конкретан текст или кључне речи
Предуслови	Текст је већ извучен и снимљен у Blackboard. Keyword Search модул је покренут како би се претрага извршила.
Опис	Корисник уноси кључну реч или фразу и систем приказује све релевантне фајлове и позиције у тексту.
Постуслови	Релевантни документи се идентификују за даљу анализу.

Основни ток	1. Корисник унесе кључну реч 2. Систем претражује Blackboard 3. Приказује се листа докумената и позиција 4. Корисник прегледа резултате
Алтернативни ток	Претрага може користити филтере по формату, датуму или језику OCR-а.

Табела 8. Претраживање извученог текста

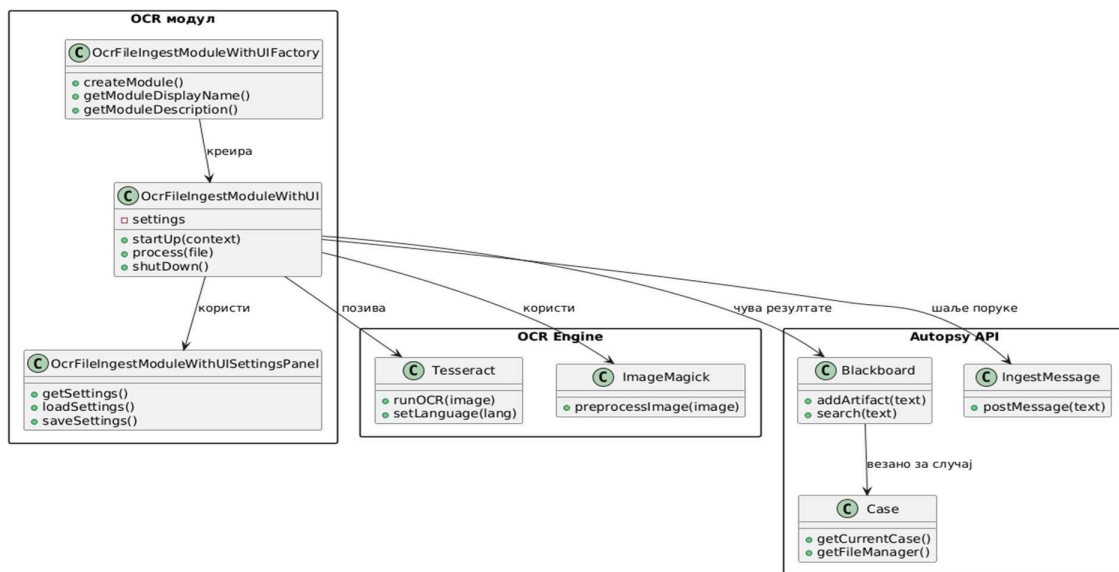
Нефункционални захтеви

Поред функционалних захтева који дефинишу шта систем треба да ради, једнако је важно и размотрити и нефункционалне захтеве, јер они одређују како систем извршава своје функције и у којим условима. У контексту развијеног OCR модула, нефункционални захтеви се односе на перформансе, стабилност, употребљивост и интеграцију у постојећи софтверски оквир. Тако се, на пример, од модула очекује да може ефикасно да обради велики број датотека у разумном времену, као и да пружи брзе и тачне резултате претраге над индексираним текстом. Истовремено, модул мора бити отпоран на грешке приликом анализе оштећених или непознатих формата фајлова, а све неправилности се морају бележити у логовима ради касније анализе.

Други значајан аспект односи се на употребљивост и компатибилност са постојећим окружењем. Интерфејс мора бити довољно једноставан да га форензичари могу користити без потребе за додатним техничким знањем, док интеграција у саму платформу мора бити без нарушавања других функционалности. Поред тога, важан захтев је безбедност података. Сав извучени текст мора се чувати унутар Autopsy базе како би се очувао интегритет доказа. На крају, модул треба да буде развијен тако да омогући лако одржавање и проширивање, како би се накнадно могли додати нови алгоритми или технике без већих измена постојеће структуре.

UML класни дијаграм

Да би се систематски приказала архитектура развијеног модула, у наставку на слици 6 је дат UML дијаграм класа. Овај дијаграм формално представља главне компоненте OCR модула, њихове атрибуте и методе, као и релације са библиотекама за обраду слика и интерфејсом Autopsy платформе. Дијаграм класа омогућава боље разумевање структуре кода и улоге појединачних класа, као и начина на који се оне међусобно повезују ради остваривања функционалности. Приказане су три централне класе модула (фабрика, модул и панел за подешавања), интеграција са OCR engine-ом као и повезаност са Autopsy API-јем кроз Blackboard, Case и IngestMessage. На овај начин обезбеђује се потпуна слика о томе како софтвер функционише у целини.



Слика 6. Класни дијаграм

Дијаграм класа приказује основну архитектуру модула и начин на који су организоване његове компоненте. Структура је постављена тако да се обезбеди јасна подела одговорности и лакше одржавање кода. Фабрика (OcrFileIngestModuleWithUIFactory) је издвојена као засебна класа јер је то стандардни образац који захтева Autopsy платформа приликом интеграције нових Python модула. Она је одговорна за креирање инстанци самог модула и омогућава његову конфигурацију. Главна функционална логика налази се у класи OcrFileIngestModuleWithUI, која управља током обраде фајлова, покреће OCR над сликама и чува резултате у форензичком

окружењу. Ова подела је неопходна јер се на тај начин раздваја кориснички интерфејс и логика рада, што олакшава проширивост и тестирање.

Класа `OcrFileIngestModuleWithUISettingsPanel` издвојена је ради конфигурације подешавања од стране корисника. Овакво решење омогућава да се модул лако прилагоди различитим сценаријима употребе, на пример избор језика за OCR или дефинисање начина обраде слика. Уместо да су подешавања интегрисана у главну класу, издвојена је посебна компонента, што омогућава бољу контролу и једноставније модификације у будућности.

Интеграција са спољним библиотекама као што су Tesseract и ImageMagick приказана је као посебан пакет „OCR Engine“. Овај приказ има за циљ да се јасно раздвоји сопствени код од зависности од библиотека. На тај начин обезбеђује се могућност замене OCR библиотека другом, без великих измена у самом модулу.

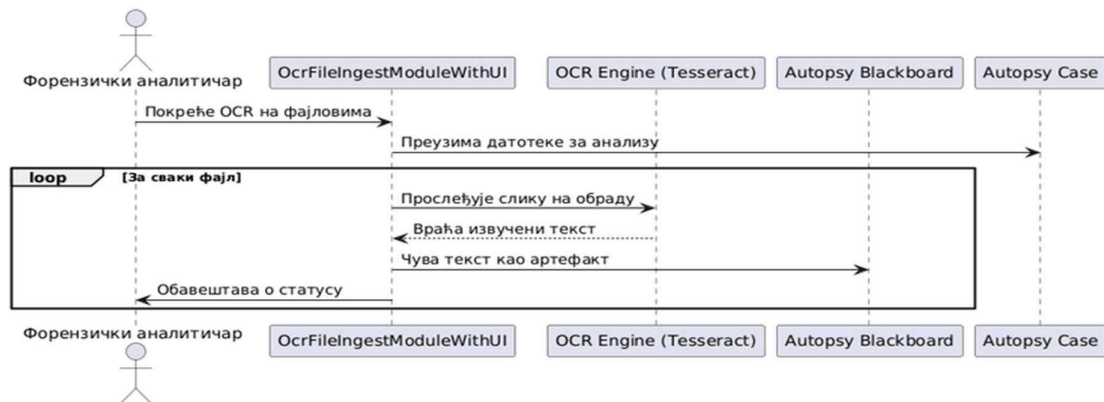
Пакет Autopsy API представља компоненте платформе са којима модул директно сарађује. Blackboard је централни део у којем се чувају артефакти (извучени текст), Case омогућава приступ активном случају и његовим датотекама, док `IngestMessage` служи за обавештавање корисника о резултатима и статусу рада модула. Ово представља јасну видљивост граница између сопствене логике модула и интерфејса који обезбеђује сама платформа.

Оваква организација представља резултат потребе за модуларношћу, лаким надоградњом и добром интеграцијом у већ постојећи екосистем Autopsy-а. Она омогућава да се модул у будућности прошири новим функционалностима (нпр. подршка за више OCR библиотека или нови типови извештаја), без значајних промена у постојећем коду.

Дијаграм секвенци

Дијаграм секвенци на слици 7 приказује ток интеракције између корисника и главних компоненти система током процеса извршавања OCR-а над датотекама. Аналитичар иницира поступак покретања модула, након чега класа `OcrFileIngestModuleWithUI` преузима датотеке из активног случаја. За сваки фајл, модул прослеђује слику прво ImageMagick engine-у за претпроцесирање. Након претпроцесирања, слика је прослеђена Tesseract OCR engine-у, који

враћа препознат текст. Добијени резултати се чувају у бази као артефакти који остају доступни у оквиру истраге. Модул кроз Autopsy интерфејс шаље информације аналитичару о статусу процеса и успешности обраде. На овај начин дијаграм секвенци приказује редослед корака и јасну поделу одговорности између различитих компоненти система.



Слика 7. Дијаграм секвенци

ИМПЛЕМЕНТАЦИЈА

Интеграција са окружењем [20]

```
class OcrFileIngestModuleWithUIFactory(IngestModuleFactoryAdapter):
    moduleName = "OCR Ingest Module (Tesseract)"

    def createFileIngestModule(self, ingestOptions):
        return OcrFileIngestModuleWithUI(self.settings)

class OcrFileIngestModuleWithUI(FileIngestModule):
    def __init__(self, settings):
        self.local_settings = settings
        self.supported_extensions = []
        self.ingestServices = IngestServices.getInstance()
        self.filesFound = 0

    def startUp(self, context):
        self.context = context
        if self.local_settings.getSetting("jpg_flag") == "true":
            self.supported_extensions.append(".jpg")
```

Листинг 1. Основна структура модула

Код приказан у листингу 1 илуструје основну структуру модула у Autopsy-ју. Фабрика (OcrFileIngestModuleWithUIFactory) је одговорна за креирање инстанце главне класе модула и његово повезивање са платформом. Autopsy платформа захтева овај образац ради регистрације сваког новог модула. Главни модул (OcrFileIngestModuleWithUI) садржи основну функционалност: чување подешавања, иницијализацију контекста и одређивање које типове фајлова модул обрађује. Метод startUp се позива пре обраде било ког фајла и служи за припрему окружења, као што је филтрирање по типу слике (JPEG, PNG, TIFF итд.).

У наставку на листингу 2 приказано је покретање претпроцесирања са ImageMagick алатом над прослеђеном датотеком.

```
def process(self, file):
    # Dynamically build the command based on user options
    magick_args = []
    magick_args.append("magick")
    magick_args.append(temp_file_path)

    # Grayscale
    if self.local_settings.getSetting("grayscale_flag") == "true":
```

```

magick_args.append("-grayscale")
magick_args.append("Rec709Luminance")

# Resize
if self.local_settings.getSetting("skip_resize_flag") == "false":
    resize_value = self.local_settings.getSetting("resize_value")
    if resize_value:
        magick_args.append("-resize")
        magick_args.append(resize_value + "%")

magick_args.append(processed_file_path)
convert_process = subprocess.Popen(magick_args,
stdout=subprocess.PIPE, stderr=subprocess.PIPE)
convert_stdout, convert_stderr = convert_process.communicate()
convert_return_code = convert_process.wait()

if convert_return_code != 0:
    self.log(Level.WARNING, "ImageMagick conversion failed for
file " + file.getName() + ". Error: " + convert_stderr)
    return IngestModule.ProcessResult.OK

self.log(Level.INFO, "Image preprocessed with ImageMagick and
saved to: " + processed_file_path)

```

Листинг 2. Препроцесирање помоћу ImageMagick алата

Унутар методе process налази се део који се бави обрадом фајлова (слика) пре него што се покрене OCR. На основу корисничких подешавања динамички се креира команда за извршење ImageMagick алата. У првој фази се одређује путања до привременог фајла оригиналне слике, а затим се додају опције за обраду слике. Корисник може да омогући grayscale опцију, која претвара слику у нијансе сиве како би OCR алгоритам препознао текст.

Друга важна опција је resize, која омогућава мењање величине слике у процентима. Ако корисник није означи да се прескаче ова опција, програм узима изабрану вредност за resize опцију и додаје је као параметар за ImageMagick. Након дефинисања свих параметара, команда се покреће спољашњи процес, а програм чека да се заврши извршење. У случају грешке у конверзији уписују се логови, док успешна конверзија резултира успешној обради слике која је након овог корака спремна за OCR обраду.

У листингу 3 приказано је подешавање Tesseract OCR алата.

```

# Run Tesseract on the processed image
tesseract_cmd = ["tesseract", processed_file_path, "stdout"]

language_code = self.local_settings.getSetting("language_code")
if language_code:
    tesseract_cmd.append("-l")
    tesseract_cmd.append(language_code)

tesseract_process =
subprocess.Popen(tesseract_cmd, stdout=subprocess.PIPE, stderr=subprocess.PIPE)
tesseract_stdout, tesseract_stderr = tesseract_process.communicate()
tesseract_return_code = tesseract_process.wait()

if tesseract_return_code != 0:
    self.log(Level.WARNING, "Tesseract failed to process file " +
file.getName() + ". Error: " + tesseract_stderr)
    return IngestModule.ProcessResult.OK

# Decode the stdout byte stream to a UTF-8 string
ocr_text = tesseract_stdout.decode('utf-8', 'ignore').strip()
if ocr_text:
    blackboard = Case.getCurrentCase().get SleuthkitCase().getBlackboard()
    attrs =
Arrays.asList(BlackboardAttribute(BlackboardAttribute.Type.TSK_KEYWORD,
OcrFileIngestModuleWithUIFactory.moduleName, ocr_text))
    art = file.newAnalysisResult(BlackboardArtifact.Type.TSK_KEYWORD_HIT,
Score.SCORE_LIKELY_NOTABLE, None, "Text Files", None, attrs).getAnalysisResult()

    blackboard.postArtifact(art, OcrFileIngestModuleWithUIFactory.moduleName,
self.context.getJobId())

```

Листинг 3. Извршење Tesseract OCR и чување артефакта у Blackboard-y.

Након претходне обраде слике, модул покреће Tesseract OCR на процесираној слици. Команда се гради динамички и укључује параметар за изабрани језик, који корисник може подесити у интерфејсу модула. Tesseract се покреће као спољашњи процес, а излазни ток процеса садржи текст са слике. Уколико извршење није успешно, програм евидентира упозорење у логовима, али наставља са обрадом других фајлова, како би се обезбедила робусност целог модула.

Након успешног извршења, излазни текст се декодира у UTF-8 стринг и резултат се анализира. Ако је пронађен текст, креира се артефакт у Autopsy Blackboard-y који садржи текст као атрибут. Артефакт се потом објављује кроз интерфејс платформе, омогућавајући кориснику да претражује и прегледа извучени текст у случају унутар Autopsy окружења.

Кориснички интерфејс (UI) модула је реализован кроз класу `OcrFileIngestModuleSettingsPanel`, која омогућава корисницима конфигурацију параметара обраде слика пре покретања OCR процеса. UI садржи групу опција за избор типова фајлова над којима се врши обрада (JPG/JPEG, PNG, TIFF и BMP) као и опције за претпроцесирање слика, попут конверзије у grayscale и resize слике. Корисници бирају да ли ће resize бити прескочен или се бира (25%, 50%, 75%, 100%).

Када слика садржи боје, OCR engine мора да обради више канала (RGB) и одвоји текст од позадине, што може утицати на тачност препознавања. Преласком у grayscale, свака тачка слике представљена је само једном вредношћу осветљености, што поједностављује слику и повећава контраст између текста и позадине. Ово смањује шум и визуелне сметње и омогућава прецизнију идентификацију облика слова и речи.

Опција resize слике представља технику којом се оригинална слика мења у мању или већу резолуцију пре самог OCR процеса. Циљ ове операције је да се оптимизује величина карактера на слици тако да буду довољно јасни за препознавање. Уколико су слова сувише мала или превише крупна, OCR engine може имати потешкоћа у тачном идентификовању симбола. Прилагођавањем резолуције обезбеђује се да слова буду у распону који је погоднији за OCR процес, што директно утиче на тачност резултата.

Опција избор језика у модулу је важна за тачно препознавање текста. Tesseract користи моделе специфичне за сваки језик. Модели садрже информације о облику слова, правопису и честим комбинацијама карактера. Ако се текст на слици не поклапа са изабраним језиком, резултати могу бити нетачни или непотпуни. Омогућавањем кориснику да бира језик модул се прилагођава конкретном језику докумената.

Све корисничке опције се динамички синхронизују са објектом `settings`, који се касније користи у методи `process` за конфигурисање ImageMagick и Tesseract позива.

Логовање омогућава кориснику праћење тока извршавања свих корака. У овом модулу логовање је реализовано коришћењем порука различитих нивоа (Info и Warning). На овај начин се обезбеђује прецизно разликовање између уобичајених обавештења и потенцијалних грешака. У листингу 4 приказан је пример чувања логова у модулу.

```

if convert_return_code != 0:
    self.log(Level.WARNING, "ImageMagick conversion failed for file " +
file.getName() + ". Error: " + convert_stderr)
    return IngestModule.ProcessResult.OK

    self.log(Level.INFO, "Image preprocessed with ImageMagick and saved to: " +
processed_file_path)

```

Листинг 4. Логовање унутар модула

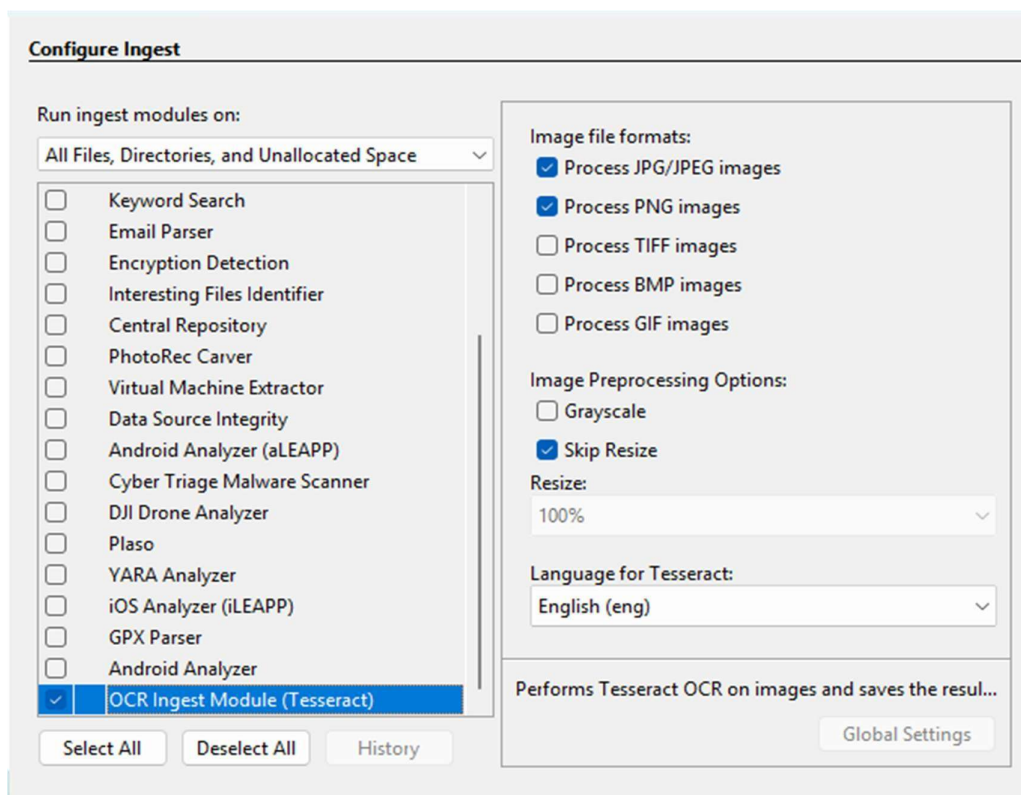
Структура кода модула је организована у складу са принципом модуларности и јасне поделе одговорности. Фабрика је одговорна за креирање инстанци модула и његово повезивање са Autopsy окружењем, главна класа управља логиком обраде, док је панел за подешавања издвојен као засебна компонента која омогућава кориснику да конфигурише начин рада.

Одрживост овог решења произилази из могућности лаке надоградње и проширивања. Зависности од спољних библиотека су јасно изоловане, што омогућава да се оне замене или ажурирају без већих измена у коду.

ДЕМОНСТРАЦИЈА

У овом поглављу приказана је употреба имплементираног OCR модула у оквиру Autopsy платформе. Циљ је демонстрација свих функционалности софтвера из угла крајњег корисника, али и да се са техничке стране анализирају добијени резултати. Демонстрација обухвата приказ корисничког интерфејса уз пратеће слике, као и неколико сценарија тестирања на реалним сликама. Поред визуелне презентације рада, анализираће се и квалитет препознавања текста, утицај различитих подешавања као и коментари и ограничења решења.

На слици 8 приказан је изглед корисничког интерфејса модула који представља подешавања за модул.



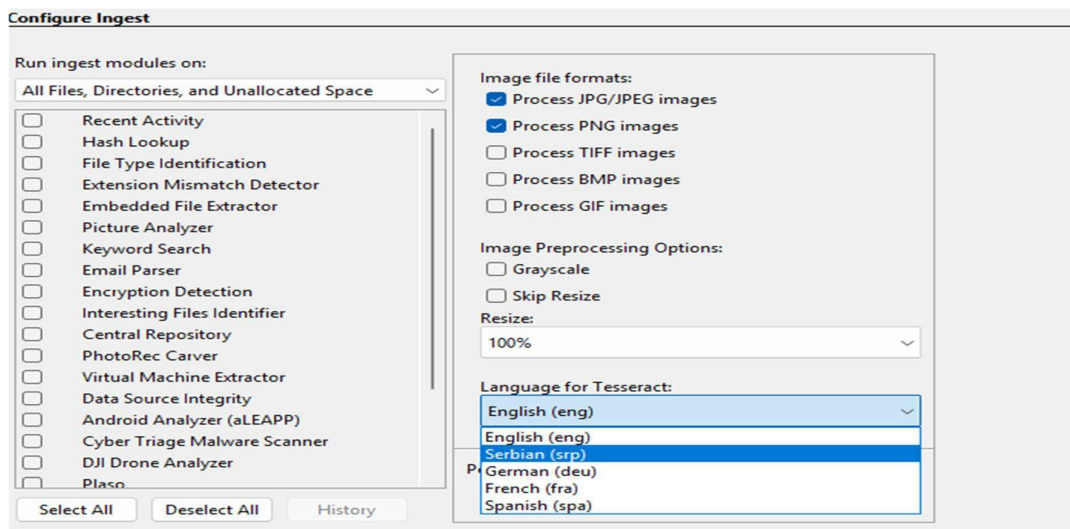
Слика 8. Кориснички интерфејс при покретању модула

Приказан је прозор за подешавање ingest модула у оквиру Autopsy платформе. Са леве стране налази се листа доступних модула који се могу укључити или искључити током ingest процеса. Међу стандардним модулима, као што су Keyword Search, Email Parser, Encryption Detection и многи други, налази се и развијени модул „OCR Ingest Module (Tesseract)“, који је чекиран ради покретања.

Десна страна прозора резервисана је за специфична подешавања изабраног модула, у овом случају OCR модула. Прва група опција односи се на типове слика фајлова које ће модул обрађивати. Корисник може означити да ли жели да процесира JPG/JPEG, PNG, TIFF, BMP или GIF формате. На приказу су укључене опције за JPG и PNG, док су остали формати неозначени.

Друга група опција односи се на препроцесирање слике пре него што се над њом изврши OCR. На располагању је претварање у сиву скалу (grayscale), као и опција да се прескочи промена величине слике (skip resize). Када је skip resize опција укључена, поље за одабир процента величине постаје онемогућено, што је и видљиво на слици као сиви падајући мени који приказује вредност 100%.

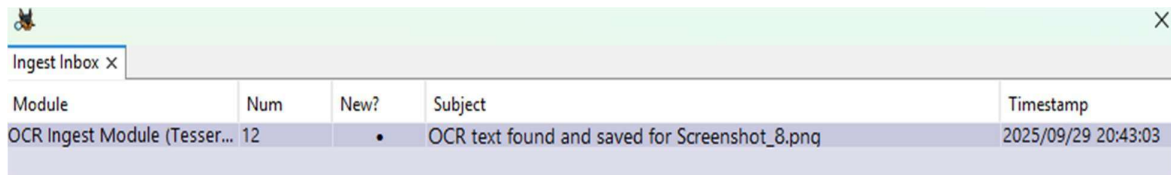
На дну подешавања налази се одабир језика за Tesseract. Падајући мени тренутно показује опцију „English (eng)“ што значи да ће се OCR анализа вршити над текстом на енглеском језику. Поред тога, испод се налази кратак опис функције модула у ком пише да модул користи Tesseract OCR како би препознао текст у сликама и сачувао резултате за даљу анализу. На слици 9 приказане су језичке опције.



Слика 9. Опције избора језика

На слици 10 је приказан прозор „Ingest Inbox“ у Autopsy платформе. Овај прозор служи као централно место за праћење рада свих покренутих ingest модула. У конкретном примеру видљиво је да модул успешно извештава о резултатима обраде. Поруке могу бити типа „info“ или „warning“, у зависности од тога да ли су поруке информативног карактера или указују на могуће

проблеме током извршавања. Овај механизам у реалном времену омогућава праћење извршења тока програма.



The screenshot shows a window titled "Ingest Inbox" with a close button (X) in the top right corner. Below the title bar is a table with five columns: "Module", "Num", "New?", "Subject", and "Timestamp". The first row of data shows "OCR Ingest Module (Tesser..." in the "Module" column, "12" in the "Num" column, a bullet point in the "New?" column, "OCR text found and saved for Screenshot_8.png" in the "Subject" column, and "2025/09/29 20:43:03" in the "Timestamp" column.

Module	Num	New?	Subject	Timestamp
OCR Ingest Module (Tesser...	12	•	OCR text found and saved for Screenshot_8.png	2025/09/29 20:43:03

Слика 10. Ingest Inbox

Поред прозора „Ingest Inbox“, важан аспект рада OCR модула представља систем логова који се генеришу током обраде сваке слике. Логови омогућавају детаљније праћење корака процеса, идентификују потенцијалне проблеме и верификују успешност обраде.

На слици 11 приказан је пример логова за једну датотеку. Види се да лог бележи сваки кључни корак од чувања привремене оригиналне слике, преко примене препроцесирања у ImageMagick алату, покретања Tesseract OCR-а, до креирања и објављивања артефаката у Blackboard. Поред тога, логови садрже информације о поставкама које је корисник конфигурисао, као што су типови обрађених слика, избор језика и примена опција препроцесирања.

Систем логова је организован тако да се поруке означавају у нивоима (INFO, WARNING и SEVERE). INFO поруке бележе успешне кораке и информације о обради, WARNING поруке сигнализирају могуће проблеме који нису довели до потпуног прекида рада, док SEVERE поруке указују на критичне грешке које спречавају нормално функционисање модула и захтевају анализу и интервенцију програмера.

```
2025-09-27 18:43:00.162 OcrFileIngestModuleWithUI process
INFO: Processing file: Screenshot_8.png, Language: ENG

2025-09-27 18:43:00.188 OcrFileIngestModuleWithUI process
INFO: Temporary original image file saved to: C:\Users\VK003\AppData\Local\Temp\tmptrf2d3.png

2025-09-27 18:43:00.81 OcrFileIngestModuleWithUI process
INFO: Image preprocessed with ImageMagick(using grayscale and resize 75%) and saved to: C:\Users\VK003\AppData\Local\Temp\tmp2_vtpl.png

2025-09-27 18:43:01.466 OcrFileIngestModuleWithUI process
INFO: OCR text extracted from Screenshot_8.png
```

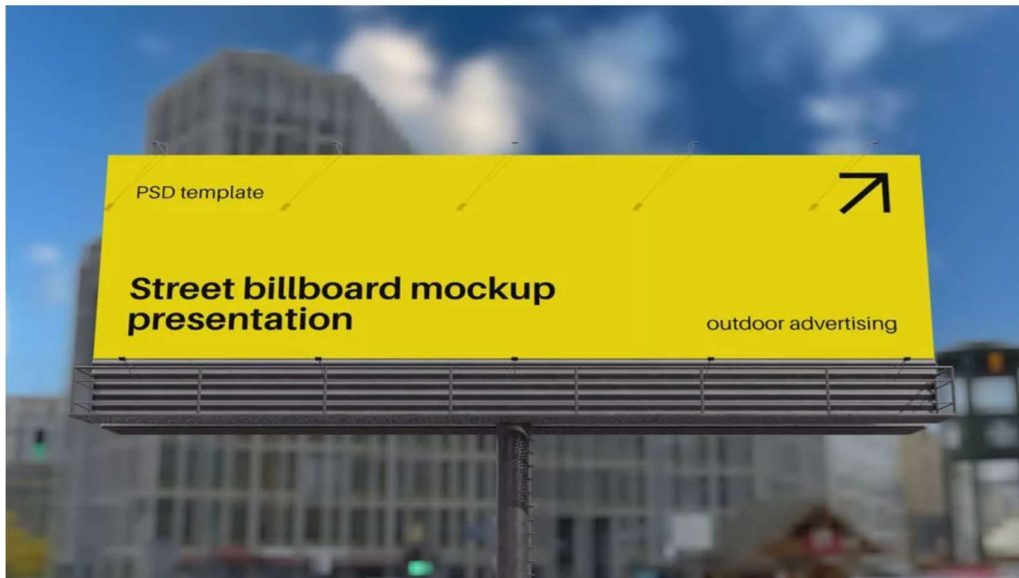
Слика 11. Приказ логова система

У наставку су приказани конкретни примери обраде слика помоћу имплементираног модула. Сваки пример ће садржати слику која је обрађена, као и резултате извученог текста. На тај начин могуће је визуелно пратити рад модула и анализирати квалитет OCR препознавања.

ПРИМЕР 1.

За први пример употребе модула коришћена су основна подешавања где су узете у обзир слике само JPG и PNG формата, grayscale и resize нису коришћени, језик је постављен на енглески.

На слици 12 приказан је жути билборд са црним текстом на сивој металној конструкцији, са замућеном градском позадином. Текст је релативно јасан и велик.



Слика 12. Жути билборд

На слици 13 приказан је резултат OCR процеса са претходно наведеним параметрима.

Data Content									
Hex	Text	Application	File Metadata	OS Account	Data Artifacts	Analysis Results	Context	Annotations	
Item: Screenshot_1.jpg									
Aggregate Score: Likely Notable									
Analysis Result 1									
Score:		Likely Notable							
Type:		Keyword Hits							
Configuration:		Text Files							
Conclusion:									
Keyword:		PSD template 7 Street billboard mockup presentation outdoor advertising							

Слика 13. Резултат анализе – жути билборд

У овом случају извлачење текста је изузетно тачно. Сви читљиви текстуални елементи, осим стрелице (која није текст) и броја „7“ (који није на слици), су правилно препознати и спојени у

један низ. Први уочени пропуст је необјашњиви додатак броја „7“ одмах након „PSD template“ у извученом тексту, с обзиром на то да тај број није визуелно присутан. Осим те грешке, текст је изузетно тачно препознат и спојен, при чему су све речи правилно извучене упркос различитим фонтовима и позицијама. Иако је модул спојио текст из више редова и поља у један низ одвојен вертикалном цртом, контекст и комплетност примарног садржаја су успешно очувани.

На слици 14 приказан је део веб странице са насловом „Најчешће постављана питања“ и два питања и одговора у вези са отварањем корисничког налога на Академској рачунарској мрежи Универзитета. Текст је ћириличан и јасно форматиран. OCR процес је извршен са претходно наведеним параметрима (пример 1).

НАЈЧЕШЋЕ ПОСТАВЉЕНА ПИТАЊА

Питање: Ко има право на отварање корисничког налога на Академској рачунарској мрежи Универзитета?

Одговор: Право на отварање корисничког налога имају сви запослени и студенти Универзитета у Новом Саду.

Питање: Шта добијам отварањем налога на Академској рачунарској мрежи?

Одговор: Отварањем налога добијате:

- e-mail адресу облика `Vaše_korisničko_ime@uns.ac.rs` уз сандуче за поруке,

- могућност коришћења других сервиса:

Слика 14. Ћирилични текст

Модул је препознао све речи, али са високим нивоом грешака у транскрипцији ћириличних слова у латиничне облике. Наравно, овакав резултат је и очекиван с обзиром да је подешавање модула постављено на енглески језик, па је модул покушао да у дугу тог језика извуче текст са слике. На слици 15 приказан је резултат.

Data Content					
Hex	Text	Application		File Metadata	
OS Account	Data Artifacts	Analysis Results	Context	Annotations	Other Occurrences
Item: Screenshot_3.png					
Aggregate Score: Likely Notable					
Analysis Result 1					
Score:	Likely Notable				
Type:	Keyword Hits				
Configuration:	Text Files				
Conclusion:					
Keyword:	HAJYELUAE NOCTABJbBEHA NUTAHbA [luTatbe: Ko uMa NpaBo Ha OTBapalbe KOPUCHUUKOr Hasora Ha AKafeMckoj pauyHapckoj MpeKu YHuBep3uTeTa? Ogrosop: NpaBo Ha OTBapatbe KOpUCHUUKOr Hasiora VMajy CBU 3anocieHu VI CTyAeHTV YHuBepsuteta y HoBom Cay. Mutare: Uta foSujam oTBapatbeM Hasiora Ha AKagfeMckoj pauyHapcKoj MpeExKU? Ogrosop: OTBapatbem Hanora Aodyjate: - e-mail agpecy o8nuka VaSe_korisniEko_ime@uns.ac.rs y3 canAyye 3a Nopyke, - MoryhHoct Kopuwihetba Apyux cepByca:				

Слика 15. Резултат анализе – ћирилични текст

Грешке:

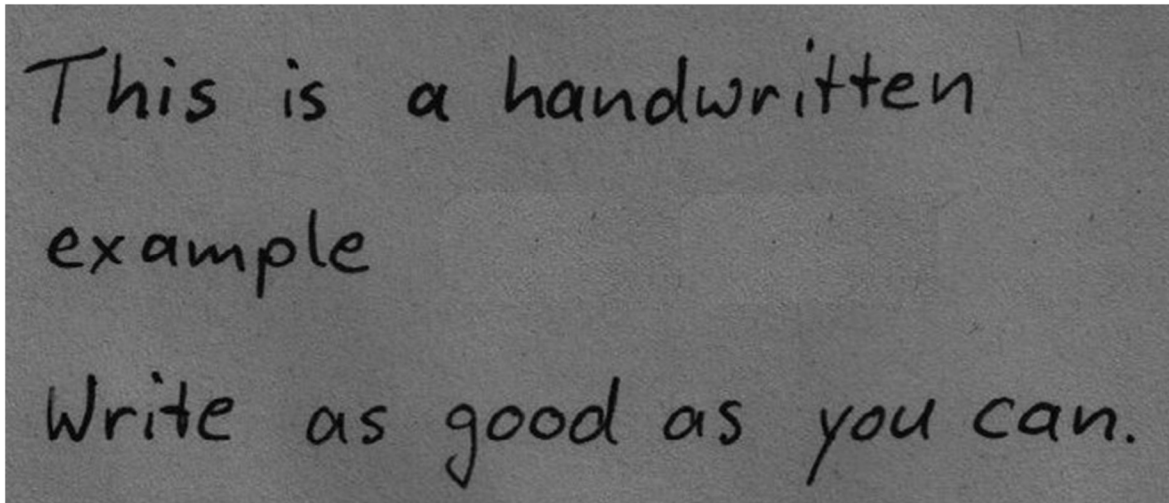
- „HAJЧЕШЋЕ“ у „HAJELUAE“ – ч и ћ су погрешно замењени симболима
- „Универзитета“ у „YHuBep3uTeTa“ – з је постало број 3, в је постало латинично В.

Грешке су конзистентне кроз цео текст, што указује да модел није био правилно подешен на српску ћирилицу, већ је користио визуелно сличне облике из другог алфабета.

Слика 16 приказује три реда рукописа на тамно сивој, благо зрнастој текстури позадине, која подсећа на папир или картонски материјал. Текст је написан црним мастилом, јасним неповезаним великим словима и садржи следеће реченице у три реда:

- „This is a handwritten“
- „Example“

- „Write as good as you can“.



Слика 16. Рукопис

Ова обрада слике које садржи јасан рукопис показала је високу ефикасност модула чак и уз коришћење основних параметара. Модул је успешно идентификовао и спојио цео текст из три реда, што указује на робустност при руковању (HTR – Handwritten Text Recognition) задатком. Сви преостали карактери, осим једне речи, су тачно извучени, при чему је цела реченична структура остала очувана.

Data Content	
Hex	Text
Application	File Metadata
OS Account	Data Artifacts
Analysis Results	Context
Annotations	
Item: Screenshot_5.png	
Aggregate Score: Likely Notable	
Analysis Result 1	
Score:	Likely Notable
Type:	Keyword Hits
Configuration:	Text Files
Conclusion:	
Keyword:	This is a handwritten example Write as goool as you can,

Слика 17. Резултат анализе - рукопис

Ипак, примећена је грешка у једној речи, што је уобичајено за преношење рукописа у дигитални текст. Реч „good“ је погрешно препозната као „gooal“ у извученом низу. Ова трансформација је највероватније резултат визуелне сличности стилизованог слова „d“ у рукопису са комбинацијом „al“. Други мањи пропуст је грешка у интерпункцији, где је модул додао зарез (,) на крају текста, уместо тачке.

Резултат је врло добар јер је цео текст, укључујући и контекст, успешно сачуван, док је једина суштинска грешка ограничена на једну измену карактера унутар једне речи. Модул је потврдио да може да обрађује рукопис и под неповољним околностима као што су тамна и текстурирана позадина.

ПРИМЕР 2.

Сет тестова у овом примеру мења улазне параметре како би се прецизније проценио утицај оптимизације слике и локализације језика на квалитет извученог текста. За разлику од претходног примера где су коришћене основне поставке, у овом сету тестова биће примењен grayscale филтер за оптимизацију контраста, док ће језик бити експлицитно постављен на српски. Циљ ове промене је да се утврди како примена ефекта сиве скале утиче на прецизност превођења текста (посебно рукописа), као и да се провери да ли експлицитно подешавање језика на српски решава проблем транскрипције који су уочени приликом обраде ћириличних карактера.

Применом овако постављених параметара на слику 14, дошло је до драстичног побољшања квалитета извученог текста. Модул је превео сав ћирилични текст у исправне ћириличне карактере. Наслов и целокупна питања и одговори су сада потпуно читљиви и тачни.

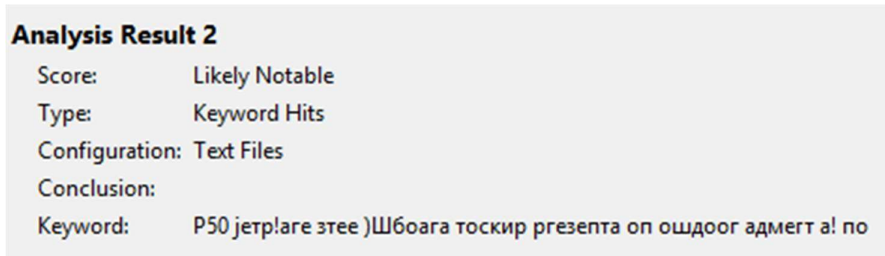
Analysis Result 2	
Score:	Likely Notable
Type:	Keyword Hits
Configuration:	Text Files
Conclusion:	
Keyword:	НАЈЧЕШЋЕ ПОСТАВЉЕНА ПИТАЊА Питање: Ко има право на отварање корисничког налога на Академској рачунарској мрежи Универзитета2 Одговор: Право на отварање корисничког налога имају сви запослени и студенти Универзитета у Новом Саду. Питање: Шта добијам отварањем налога на Академској рачунарској мрежи Одговор: Отварањем налога добијате: - е-та! адресу облика Мазе_Ккоп5пско [теФип5.ас.г5 уз сандуче за поруке, - могућност коришћења других сервиса:

Слика 18. Резултат анализе – ћирилични текст (језик српски)

Упоређујући резултате са слике 15 и слике 18, јасно је да је експлицитно подешавање језика на српски решило огромне проблеме из резултата првог случаја. Иако је grayscale могао да помогне у оптимизацији главни фактор побољшања је била коректна локализација. У новом резултату, преостале грешке су:

- Замена знака упитник (?) бројевима 2 и г на крајевима питања.
- Парцијално неправилно препознавање у делу е-mail адресе (која је латинична). Ово сугерише да, иако је ћирилица побољшана, прелазак на латиницу унутар ћириличног текста и даље може представљати изазов.

На слици 19 приказан је резултат процесирања слике 12 (жути билборд). Као што је дошло до грешке у извлачењу латиничног текста током ћириличног текста, тако долази до истог проблема и у случају енглеског језика са српским ћириличним писмом.



Слика 19. Резултат анализе – жути билборд (језик српски)

У овом случају, где је на слици латинични текст, постављање језика на српски довело је до значајног погоршања резултата. Модул је погрешно превео једноставну латиницу у низ бесмислених знакова, док је под енглеским језиком у примеру 1 текст био готово препознат без грешке.

ПРИМЕР 3.

Утицај смањења резолуције и grayscale филтера на прецизност OCR модула. За ову анализу, за слику 12 је употребљена resize опција на 75% оригиналне величине и примењен је Grayscale филтер, док је коришћен језик енглески. Извучени текст је видљив на слици 20.

Analysis Result 7

Score: Likely Notable
Type: Keyword Hits
Configuration: Text Files
Conclusion:
Keyword: PSD template Street billboard mockup presentation outdoor advertising

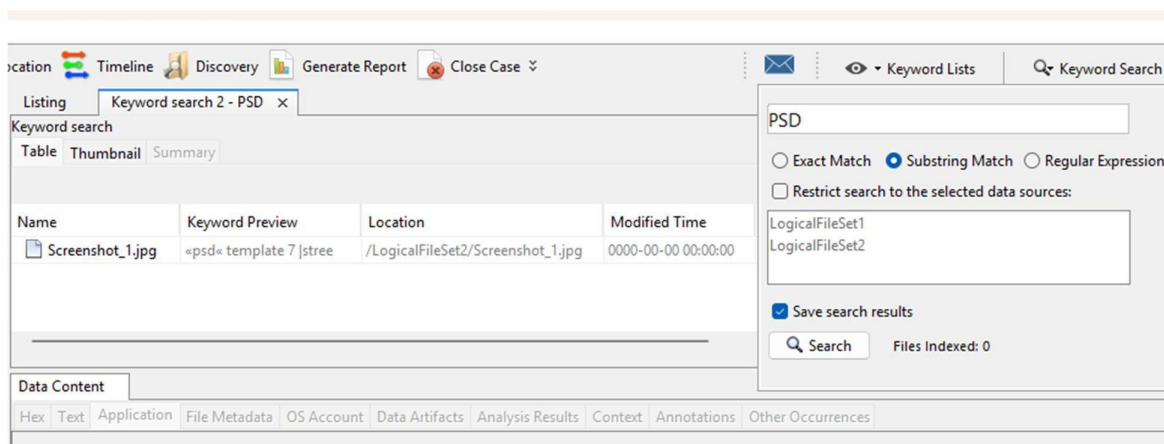
Слика 20. Резултат анализе – жути билборд (resize 75%, grayscale)

Првобитна грешка у примеру 1 на слици 13 била је „халуцинација“ броја „7“. Применом resize 75% опције и grayscale опције, постигнут је бољи резултат у односу на претходни случај. Извучени текст приказује највиши степен тачности. Кључна побољшања постигнута су комбиновањем филтера, при чему је смањење резолуције уклонило визуелни шум одговоран за халуцинацију броја 7, док је повећан контраст омогућио прецизније извлачење истакнутог текста.

ПРИМЕР 4.

Сваки текст из сваке анализе је сачуван и индексиран унутар Autopsy Blackboard-a. Демонстрирано је како се може искористити претрага по кључним речима за проналажење специфичних информација унутар дигитализованог садржаја. Ова функција је кључна за ефикасно управљање великим количинама текста из слика и показује практичну примену OCR технологије у анализи података.

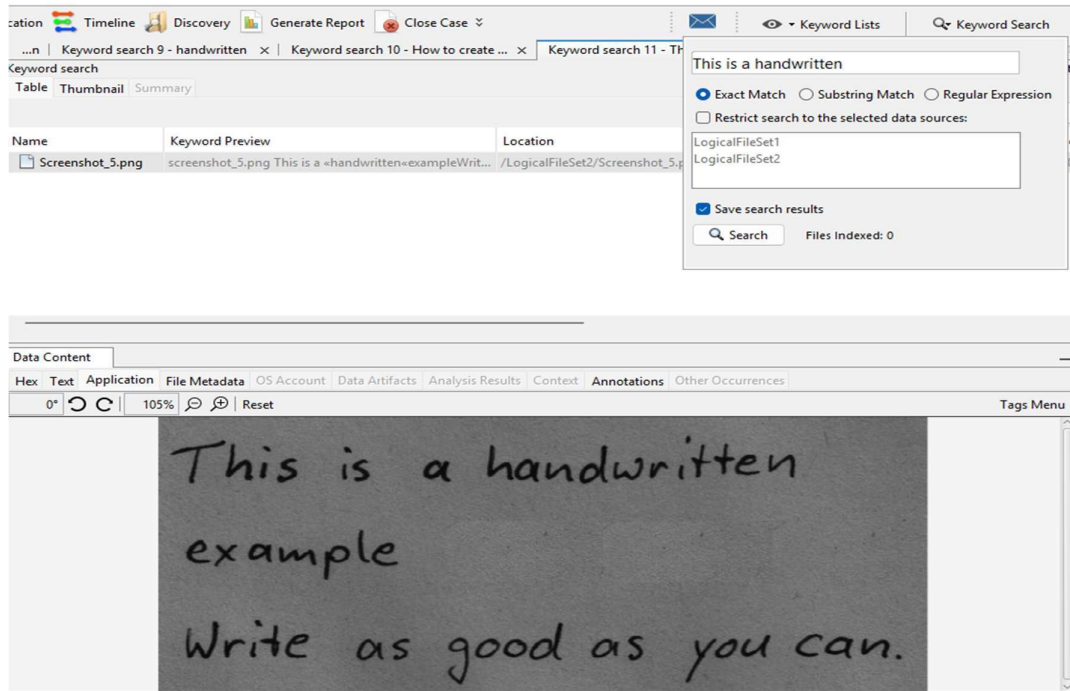
На слици 21 је приказан резултат претраге по кључним речима употребом постојећег Keyword Search модула.



Слика 21. Резултат претраге по кључној речи

На слици жути билборд (слика 12), налази се кључна реч „PSD“ коју је након извршења развијеног OCR модула могуће претраживати. Резултат претраге приказује слику у којој се кључна реч налази. Индексирање у Autopsy Blackboard је извршено као keyword hit, што омогућава претраживање по кључним речима.

Пример претраге по фрази приказан је на слици 22 где се може видети резултат претраге.



Слика 22. Резултат претраге по фрази

ЗАКЉУЧАК

У овом раду анализирана је потреба за аутоматизованом обрадом и препознавањем текста из дигиталних слика у контексту форензичких истрага. Основни проблем који је идентификован односи се на потешкоће које се јављају при ручном прегледу великих количина слика у потрази за корисним информацијама, што је дуготрајно и подложно људској грешци. За решавање овог проблема развијен је OCR модул заснован на Tesseract OCR-у који је интегрисан у платформу Autopsy. Мотивација за реализацију овог модула била је да се обезбеди ефикасан алат који омогућава брзо, поуздано и флексибилно извлачење текста из различитих формата слика, што директно доприноси квалитету форензичке анализе.

Специфицирано решење и реализовани модул доносе неколико значајних предности у односу на слична постојећа решења. Пре свега, интеграција у Autopsy платформу омогућава да корисник из једног окружења управља целокупним процесом обраде и анализе слика, без потребе за додатним алатима. Кориснички интерфејс је интуитиван и пружа опције за избор формата слика, подешавања језика, могућност опција препроцесирања, што повећава тачност и применљивост модула. Модул нуди и напредне аналитичке могућности као што су препознавање рукописног текста и контекстуална анализа текста. Такође, логовање процеса и структурирани код омогућавају лако праћење корака обраде, дијагностику грешака и одрживост кода у будућности. Све ове карактеристике чине модул флексибилним алатом погодним за разне случајеве.

Ипак, имплементирани модул има и своја ограничења. OCR резултати могу бити непоуздани у случајевима када су слике ниске резолуције, имају сложену позадину или текст није у потпуности подржан од стране изабраног језика. Алат такође не може сам да изабере параметре за претходну обраду, као што су промена величине слике и избор језика, што представља додатни правац за будући развој. Обрада великих или бројних слика захтева значајне хардверске ресурсе, посебно када су укључене операције препроцесирања, што може успорити процес анализе у већим истрагама. У поређењу са другим решењима, модул је фокусиран само на слике. Ово представља простор за побољшање у наредним верзијама.

За будући развој OCR модула могу се предложити неколико правца. Оптимизација алгоритама за рад са великим сликама и повећање ефикасности процеса препроцесирања може значајно смањити време обраде и повећати практичност модула у реалним условима. Проширење

подршке за више језика, интеграција напредних аналитичких алата за обрађени текст као што су категоризација, семантичка анализа и дубља контекстуална анализа. Категоризација подразумева аутоматско груписање текста по темама или типовима информација, семантичка анализа омогућава разумевање значења речи и њихових односа у тексту, а дубља контекстуална анализа омогућава извлачење сложених информација узимајући у обзир шири контекст докумената. Све ове надоградње учиниле би модул још кориснијим у форензичким истрагама и истраживању дигиталних доказа.

БИБЛИОГРАФИЈА

- [1] The Sleuth Kit, Brian Carrier [Online]. Доступно: <https://www.sleuthkit.org/sleuthkit/>, [Приступљено: 24. Септембар, 2025]
- [2] Autopsy, History [Online]. Доступно: <https://www.sleuthkit.org/autopsy/history.php> [Приступљено: 24. Септембар, 2025]
- [3] sleuthkit/autopsy github repository, [Online]. Доступно: <https://github.com/sleuthkit/autopsy> [Приступљено: 24. Септембар, 2025]
- [4] Autopsy User Documentation [Online]. Доступно: <https://sleuthkit.org/autopsy/docs/user-docs/4.18.0/> [Приступљено: 24. Септембар, 2025]
- [5] Download Autopsy [Online]. Доступно: <https://www.autopsy.com/download/> [Приступљено: 24. Септембар, 2025]
- [6] Autopsy modules [Online]. Доступно: https://sleuthkit.org/autopsy/docs/api-docs/4.19.3/platform_page.html [Приступљено: 25. Септембар, 2025]
- [7] Ingest modules [Online]. Доступно: https://sleuthkit.org/autopsy/docs/user-docs/4.0/ingest_page.html [Приступљено: 25. Септембар, 2025]
- [8] Keyword Search module [Online]. Доступно: https://sleuthkit.org/autopsy/docs/user-docs/4.0/keyword_search_page.html [Приступљено: 25. Септембар, 2025]
- [9] EXIF Parser Module [Online]. Доступно: https://sleuthkit.org/autopsy/docs/user-docs/4.0/exif_parser_page.html [Приступљено: 25. Септембар, 2025]
- [10] Email Parser Module [Online]. Доступно: https://sleuthkit.org/autopsy/docs/user-docs/4.0/email_parser_page.html [Приступљено: 25. Септембар, 2025]
- [11] File Type Identification Module [Online]. Доступно: https://sleuthkit.org/autopsy/docs/user-docs/4.0/file_type_identification_page.html [Приступљено: 25. Септембар, 2025]

- [12] Java module development [Online]. Доступно: https://sleuthkit.org/autopsy/docs/api-docs/4.19.3/mod_dev_page.html [Приступљено: 25. Септембар, 2025]
- [13] Python module development [Online]. Доступно: https://sleuthkit.org/autopsy/docs/api-docs/4.19.3/mod_dev_py_page.html [Приступљено: 25. Септембар, 2025]
- [14] EnCase [Online]. Доступно: <https://www.opentext.com/products/forensic> [Приступљено: 25. Септембар, 2025]
- [15] FTK Forensic Toolkit [Online]. Доступно: <https://www.exterro.com/digital-forensics-software/forensic-toolkit> [Приступљено: 25. Септембар, 2025]
- [16] X-Ways Forensics [Online]. Доступно: <https://www.x-ways.net/forensics/> [Приступљено: 25. Септембар, 2025]
- [17] What is OCR? [Online]. Доступно: <https://www.ibm.com/think/topics/optical-character-recognition> [Приступљено: 26. Септембар, 2025]
- [18] What is Optical Character Recognition? OCR explained by Google [Online]. Доступно: <https://cloud.google.com/blog/products/ai-machine-learning/what-is-ocr> [Приступљено: 26. Септембар, 2025]
- [19] A comprehensive evaluation of TrOcr [Online]. Доступно: https://nhsjs.com/2024/a-comprehensive-evaluation-of-trocr-with-varying-image-effects/?utm_source [Приступљено: 26. Септембар, 2025]
- [20] Github репозиторијум модула [Online]. Доступно: <https://github.com/nemanjaa21/OCR-Ingest-Module> [Приступљено: 27. Септембар, 2025]
- [21] ABBY Fine Reader [Online]. Доступно: <https://pdf.abbyy.com/> [Приступљено: 28. Септембар, 2025]
- [22] Microsoft Azure OCR [Online]. Доступно: <https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/overview-ocr> [Приступљено: 28. Септембар, 2025]

- [23] Amazon Textract [Online]. Доступно: <https://aws.amazon.com/textract/> [Приступљено: 28. Септембар, 2025]
- [24] Handwriting OCR [Online]. Доступно: <https://www.handwritingocr.com/> [Приступљено: 28. Септембар, 2025]
- [25] OCR and Indexing [Online]. Доступно: <https://support.filevine.com/hc/en-us/articles/360034968272-OCR-and-Indexing> [Приступљено: 28. Септембар, 2025]
- [26] C.M. Bishop, Pattern recognition and machine learning. New York, NY, USA: Springer, 2006. [Online]. Доступно: <https://github.com/Benlau93/Data-Science-Curriculum/blob/master/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf> [Приступљено: 28. Септембар, 2025]

БИОГРАФИЈА



Немања Малиновић је рођен 2000. године у Новом Саду. Завршио је средњу школу ЕТШ „Михајло Пупин“ у Новом Саду 2019. године, смер електротехничар рачунара. Исте године уписао је основне академске студије на Факултету техничких наука у Новом Саду, студијски програм примењено софтверско инжењерство. Мастер академске студије уписао је 2023. године на Факултету техничких наука у Новом Саду, студијски програм рачунарство и аутоматика, смер примењене рачунарске науке и информатика, модул Електронско пословање. Положио је све испите прописане планом и програмом и испунио је све услове за одбрану завршног рада.