

**Sistem de Watermarking  
pentru Adăugarea și Detectarea  
Watermark-urilor în Imagini**

**Autor:** Ciuculan Ștefan

**An:** 2024

# Cuprins

<b>Introducere .....</b>	<b>3</b>
<b>Tema Proiectului .....</b>	<b>3</b>
<b>Obiectivele Proiectului .....</b>	<b>4</b>
<b>Realizări similare .....</b>	<b>4</b>
<b>Etapă de realizare și implementare .....</b>	<b>5</b>
<b>Aspecte semnificative legate de AM .....</b>	<b>6</b>
<b>Utilitatea aplicației în viața cotidiană .....</b>	<b>6</b>
<b>Prezentare a modului de utilizare .....</b>	<b>7</b>
<b>Concluzii .....</b>	<b>10</b>
<b>Bibliografie .....</b>	<b>11</b>

## **Introducere**

Sistemul de inserare și detectare a watermark-urilor în imagini, denumit "ThirstyMark", este o aplicație dezvoltată folosind Python și modulele sale pentru procesare de imagini și interfață grafică. Scopul principal al acestui sistem este de a permite utilizatorilor să adauge watermark-uri textuale sau logo-uri pe imagini, în scopul protejării drepturilor de autor sau a identificării proprietarului imaginii. De asemenea, sistemul oferă și funcționalitatea de detectare a watermark-urilor, pentru a identifica și evidenția aceste elemente într-o imagine.

## **Tema**

Tema proiectului este axată pe utilizarea watermark-urilor pentru identificarea și protejarea proprietății intelectuale în imagini. Watermark-urile sunt esențiale în acest context, deoarece oferă o modalitate eficientă de a marca și de a proteja lucrările artistice și fotografice împotriva utilizării neautorizate sau a manipulării.

Principiul de bază al watermark-urilor este acela de a integra informații suplimentare în imagine într-un mod care să fie vizibil, dar care să nu afecteze în mod semnificativ calitatea sau aspectul acesteia. Aceste informații adiționale pot fi sub formă de texte, logo-uri sau alte elemente grafice care identifică proprietarul sau creatorul imaginii.

Utilizarea watermark-urilor poate servi la mai multe scopuri importante în protejarea proprietății intelectuale:

1. **Protecția împotriva utilizării neautorizate:** Watermark-urile pot descuraja utilizarea neautorizată a imaginilor, deoarece marchează clar proprietatea intelectuală și indică drepturile de autor ale proprietarului. Acest lucru face mai dificilă sau imposibilă utilizarea sau distribuirea imaginilor fără permisiune.

2. **Identificarea proprietarului sau creatorului imaginii:** Watermark-urile pot include informații precum numele proprietarului, logo-ul sau alte detalii de identificare care să asocieze clar imaginea cu creatorul său. Acest lucru este util în cazul în care imaginea este distribuită sau utilizată în alte contexte, deoarece permite identificarea rapidă a proprietarului și stabilirea drepturilor de autor.

3. **Dovada drepturilor de autor:** Watermark-urile pot servi ca dovadă a drepturilor de autor în cazul unor dispute legale sau litigii referitoare la utilizarea neautorizată a imaginilor. Prezența clară a unui watermark poate constitui o evidență puternică a proprietății intelectuale și poate sprijini cazul proprietarului în fața instanței.

4. **Creșterea vizibilității și a conștientizării brandului:** Plasarea unui logo personalizat sau a unor elemente grafice distinctive pe imagini poate contribui la creșterea vizibilității și a conștientizării brandului în mediul online. Fiecare imagine distribuită devine astfel o formă de publicitate subtilă, consolidând identitatea vizuală a brandului și atrăgând atenția asupra acestuia.

Sistemul "ThirstyMark" abordează două aspecte principale legate de watermark-uri:

1. **Inserarea Watermark-urilor:** Utilizatorii pot adăuga watermark-uri textuale sau logo-uri pe imagini, având opțiunea de a personaliza textul sau a-și adăuga propriul logo.

2. **Detectarea Watermark-urilor:** Sistemul poate identifica și evidenția watermark-urile existente într-o imagine, ajutând utilizatorii să le recunoască sau să le gestioneze.

### **Obiectivele**

#### **Obiective Principale:**

1. Dezvoltarea unei aplicații intuitive și ușor de folosit pentru inserarea și detectarea watermark-urilor în imagini.
2. Implementarea funcționalităților de adăugare a watermark-urilor textuale și grafice.
3. Integrarea unei funcționalități robuste de detectare a watermark-urilor, utilizând tehnici de prelucrare a imaginilor și algoritmi de segmentare.

#### **Obiective Secundare:**

1. Asigurarea unei interfețe grafice plăcute și responsive, care să faciliteze interacțiunea utilizatorului cu sistemul.
2. Optimizarea performanței aplicației pentru manipularea eficientă a imaginilor de diferite dimensiuni și complexități.

### **Realizări Similare**

Există mai multe soluții software și framework-uri disponibile care oferă funcționalități similare pentru inserarea și detectarea watermark-urilor în imagini. Iată câteva exemple notabile:

#### **1. Adobe Photoshop:**

- Este unul dintre cele mai cunoscute și utilizate programe de editare foto și grafică din lume.
- Funcționalitatea sa extinsă permite utilizatorilor să adauge watermark-uri textuale sau grafice în mod flexibil și să le ajusteze pentru a se potrivi nevoilor lor specifice.
- Photoshop oferă o varietate de instrumente avansate de design și editare, permițând utilizatorilor să creeze watermark-uri complexe și estetice.
- Capacitatea sa de a lucra cu straturi (layers) și de a gestiona transparența permite utilizatorilor să integreze watermark-urile în mod eficient în imagini fără a compromite calitatea acestora.

#### **2. OpenWPM:**

- Este un framework open-source dezvoltat în Python, specializat în manipularea și gestionarea watermark-urilor în imagini.
- Oferă funcționalități cuprinzătoare pentru adăugarea, gestionarea și detectarea watermark-urilor, cu un accent deosebit pe automatizare și scalabilitate.
- OpenWPM utilizează tehnologii web și Python pentru a oferi o platformă versatilă și puternică pentru gestionarea watermark-urilor într-un mod eficient și flexibil.
- Capacitatea sa de a fi personalizat și extins face din OpenWPM o opțiune populară pentru proiecte care necesită o soluție robustă și adaptabilă pentru gestionarea watermark-urilor în imagini.

#### **3. Stegano:**

- Este o bibliotecă Python specializată în steganografie, adică ascunderea informațiilor în imagini, care poate fi folosită și pentru adăugarea de watermark-uri.
- Oferă o gamă largă de tehnici și algoritmi de steganografie, inclusiv metode pentru inserarea și extragerea de watermark-uri în imagini.
- Stegano este flexibil și ușor de utilizat, oferind utilizatorilor posibilitatea de a experimenta cu diferite tehnici și algoritmi pentru a găsi cea mai potrivită soluție pentru nevoile lor specifice.

#### 4. **Digital Invisible Ink Toolkit (DIIT):**

- Este un set de instrumente dezvoltat în cadrul Laboratorului de Inteligență Artificială din Stanford, specializat în inserarea și detectarea de watermark-uri digitale.
- DIIT oferă funcționalități avansate pentru securitatea informațiilor și protejarea drepturilor de autor, utilizând tehnologii moderne de criptografie și inteligență artificială.
- Este conceput pentru a fi utilizat într-un mediu profesional și academic, oferind cercetătorilor și dezvoltatorilor o platformă puternică și versatilă pentru experimentarea și implementarea de tehnologii de watermarking avansate.

### **Prezentare Tehnică a Etapei de Realizare/Implementare**

#### **Biblioteci și Module Utilizate:**

- **Tkinter:** Folosit pentru dezvoltarea interfeței grafice a aplicației, inclusiv butoane, câmpuri de text și canvas-ul pentru afișarea imaginilor.
- **Pillow (Python Imaging Library):** Utilizat pentru manipularea imaginilor, inclusiv adăugarea de text și watermark-uri, redimensionarea și salvarea imaginilor.
- **OpenCV:** Folosit pentru operațiuni avansate de procesare a imaginilor, cum ar fi conversia la scala de gri, segmentarea și detectarea conturilor.

#### **Algoritmi și Metodologii:**

##### 1. **open\_image(self):**

- Această metodă este responsabilă pentru deschiderea unei imagini selectate de către utilizator.
- Utilizează **filedialog.askopenfilename()** pentru a deschide o fereastră de dialog și a permite utilizatorului să selecteze o imagine.
- Imaginea este încărcată folosind **Image.open(image\_path)**.
- Este redimensionată pentru a se potrivi dimensiunilor canvas-ului dacă este necesar.
- Imaginea este afișată pe canvas folosind **self.canvas.create\_image()**.

##### 2. **add\_text(self):**

- Această metodă adaugă text ca watermark pe imagine.
- Primește textul din **self.text\_area**.
- Creează o imagine nouă cu fundal transparent pentru a desena textul pe ea.
- Desenează textul folosind **ImageDraw.text()** pe imaginea de fundal transparentă.

- Așează imaginea cu text pe imaginea originală și pe cea folosită pentru previzualizare.
  - Actualizează canvas-ul cu noua imagine care conține watermark-ul de text.
3. **add\_logo(self):**
    - Această metodă permite utilizatorului să adauge un logo pe imagine.
    - Folosește **filedialog.askopenfilename()** pentru a selecta un logo.
    - Redimensionează logo-ul dacă este necesar.
    - Plasează logo-ul pe imaginea originală și pe cea folosită pentru previzualizare.
    - Actualizează canvas-ul cu noua imagine care conține logo-ul adăugat.
  4. **save\_image(self):**
    - Salvează imaginea modificată după adăugarea watermark-ului sau a logo-ului.
    - Utilizează **filedialog.asksaveasfilename()** pentru a permite utilizatorului să selecteze locația și numele fișierului salvat.
    - Salvează imaginea folosind **Image.save()**.
  5. **detect\_watermark(self):**
    - Această metodă detectează watermark-urile în imagine utilizând contururile.
    - Convertește imaginea din canvas într-un array numpy.
    - Convertește imaginea în tonuri de gri folosind **cv2.cvtColor()**.
    - Aplică un prag fix pentru a segmenta imaginea folosind **cv2.threshold()**.
    - Găsește contururile în imaginea segmentată folosind **cv2.findContours()**.
    - Desenează contururile pe imaginea originală.
    - Actualizează canvas-ul cu imaginea care conține contururile.

### **Aspectele Semnificative Legate de Aplicații Multimedia (AM):**

- **Manipularea Imaginilor:** Aplicația se concentrează pe manipularea și procesarea imaginilor, inclusiv adăugarea și detectarea watermark-urilor.
- **Interfață Grafică Intuitivă:** Utilizatorii interacționează cu sistemul printr-o interfață grafică simplă și intuitivă, care facilitează încărcarea, vizualizarea și salvarea imaginilor.

### **Contribuția Personală:**

- Dezvoltarea codului sursă pentru implementarea funcționalităților de adăugare a watermark-urilor textuale și logo-urilor.
- Integrarea algoritmilor de detectare a watermark-urilor folosind biblioteca OpenCV.
- Proiectarea și implementarea interfeței grafice pentru a oferi o experiență utilizator plăcută și eficientă.

### **Utilitate în diverse domenii din viața cotidiană**

1. **Protecția drepturilor de autor pentru fotografi și artiști:** Fotografiile și operele de artă digitală sunt adesea expuse riscului de utilizare neautorizată sau piratare pe internet. Prin adăugarea unui watermark personalizat, creatorii pot proteja lucrările lor și pot evidenția drepturile de autor.

2. **Promovarea brandului:** Companiile și brandurile pot utiliza watermark-urile pentru a-și promova identitatea vizuală pe platformele de social media sau în materialele de marketing. Adăugarea unui logo sau a numelui brandului pe imagini poate consolida recunoașterea brandului și poate crește vizibilitatea online.

3. **Prevenirea furtului de conținut online:** Bloggerii, influencerii și creatorii de conținut pot folosi watermark-urile pentru a împiedica furtul de conținut și pentru a proteja munca lor creativă. Acest lucru poate descuraja utilizatorii să copieze și să distribuie conținutul fără permisiunea adecvată.

4. **Identificarea proprietarului imaginii:** În cazul în care o imagine este distribuită pe scară largă pe internet sau în alte medii, watermark-ul poate servi drept mijloc de identificare a proprietarului sau creatorului imaginii. Acest lucru poate fi util în situațiile în care este necesară autentificarea sau evidențierea autorului original al unei imagini.

5. **Prevenirea utilizării neautorizate în proiecte comerciale:** Profesioniștii din industria creativă, cum ar fi în industria designului grafic sau a fotografiei, pot folosi watermark-urile pentru a proteja lucrările lor și pentru a evita utilizarea neautorizată în proiecte comerciale sau publicitare fără acordul lor.

## **Prezentare a Modulului de Utilizare**

### **Interacțiune cu Utilizatorul:**

#### **1. Deschiderea unei Imagini:**

- Utilizatorul poate deschide o imagine folosind butonul "Open Image". (Fig. 1) I se va deschide o fereastră de navigare printre fișierele sistemului pentru a căuta imaginea pe care se dorește a se lucra. Odată ce imaginea este încărcată, aceasta va fi afișată pe canvas-ul din partea stângă a aplicației.

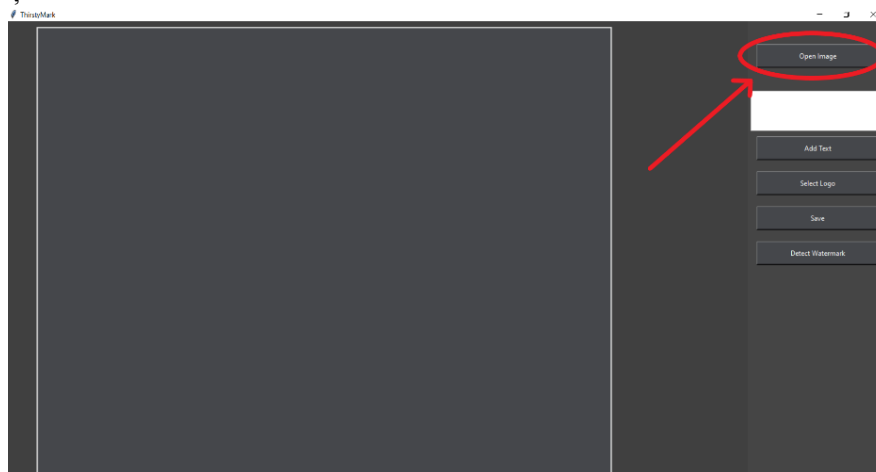


Fig.1

#### **2. Adăugarea Watermark-ului Textual:**

- După ce imaginea este încărcată, utilizatorul poate introduce textul dorit în caseta de text și apoi poate apăsa butonul "Add Text" pentru a adăuga watermark-ul text pe imagine. Pe

canvas-ul din stânga va fi adăugat watermark-ul textual ca o previzualizare pentru a observa cum va arăta imaginea după salvare. (Fig. 2)

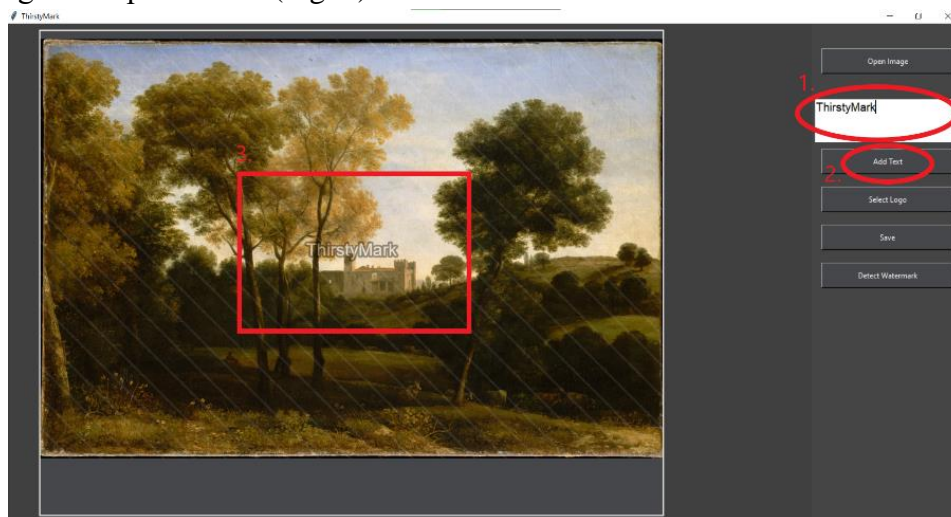


Fig. 2

### 3. Adăugarea Watermark-ului Grafic (Logo):

- Utilizatorul poate selecta un logo de la calculatorul său apăsând butonul "Select Logo". (Fig. 3, Fig. 4) Acesta va fi plasat în colțul din stânga jos al imaginii. (Fig. 5)

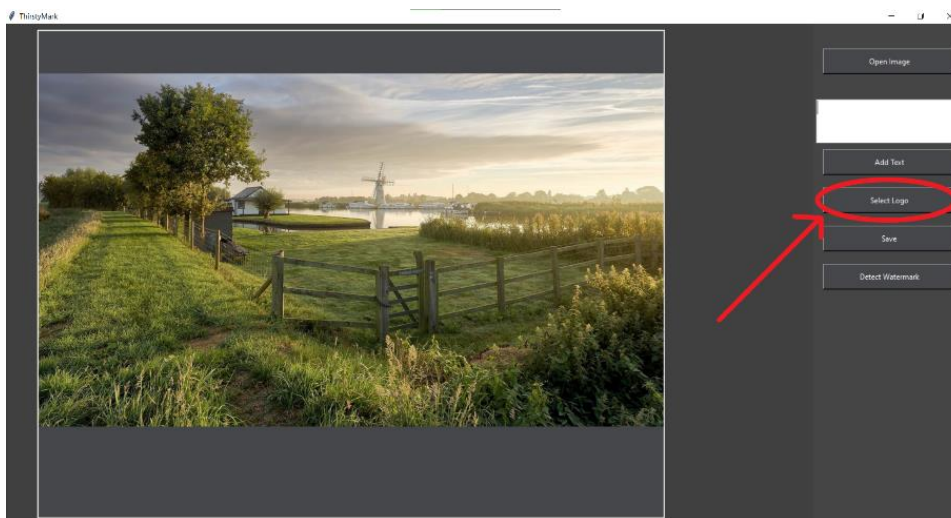


Fig.3



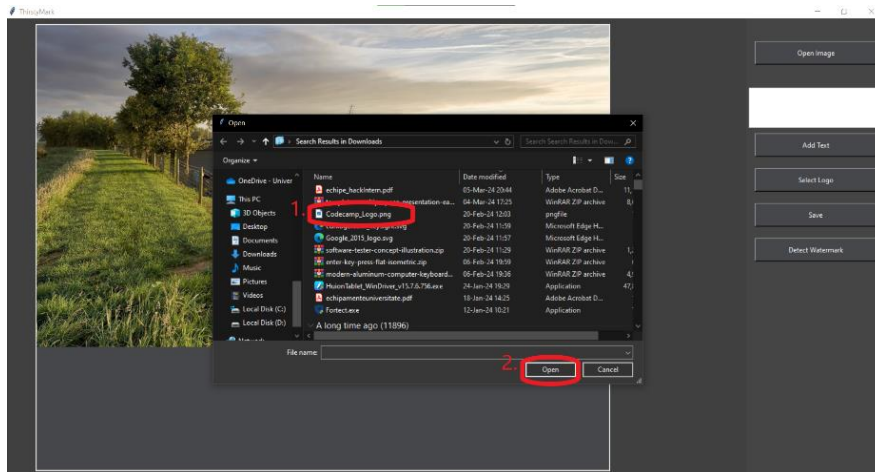


Fig. 4



Fig. 5

#### 4. Salvarea Imaginii Modificate:

- După ce watermark-ul este adăugat și utilizatorul este mulțumit de modificări, poate salva imaginea modificată apăsând butonul "Save". (Fig. 6)



Fig. 6

- Odată ce imaginea este salvată, copia în miniatură de pe canvas este ștearsă pentru a nu încărca prea mult spațiul din memorie.

#### 5. Detectarea Watermark-ului:

- Utilizatorul poate apăsa butonul "Detect Watermark" (Fig. 7) pentru a evidenția watermark-urile detectate în imaginea încărcată. (Fig. 8)

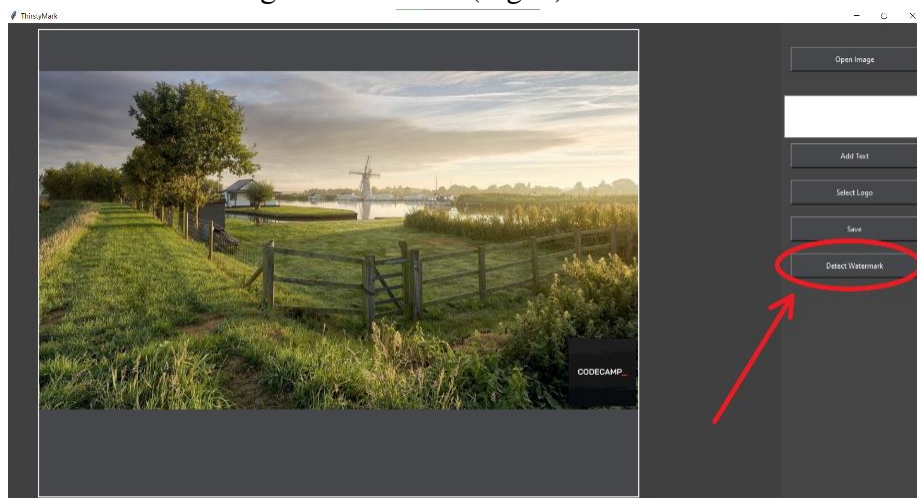


Fig. 7

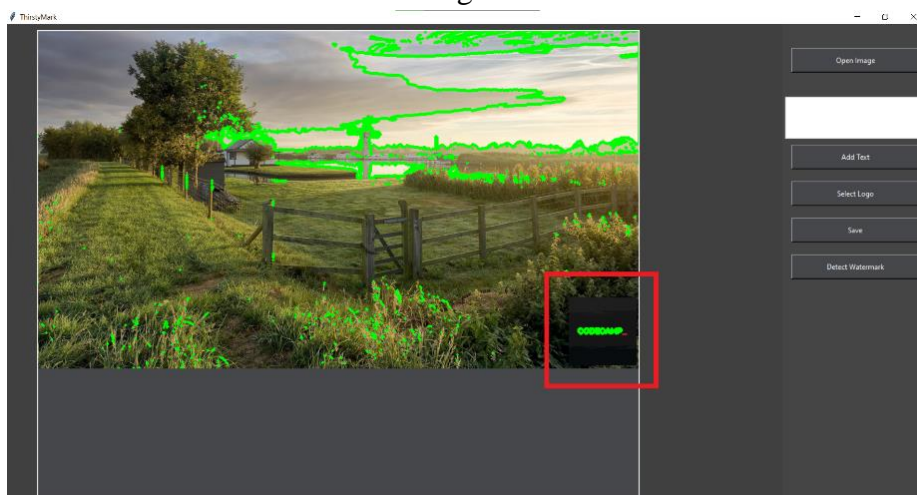


Fig.8

### Concluzii

#### Îndeplinirea Obiectivelor:

- **Dezvoltarea unei Aplicații Intuitive:** Aplicația "ThirstyMark" oferă o interfață grafică simplă și intuitivă, care permite utilizatorilor să încarce, să adauge watermark-uri și să detecteze watermark-uri în imagini cu ușurință.
- **Implementarea Funcționalităților Specifice:** Toate funcționalitățile propuse, cum ar fi adăugarea de watermark-uri textuale și grafice, detectarea watermark-urilor și salvarea imaginilor modificate, au fost implementate cu succes și funcționează conform așteptărilor, cu

câteva excepții în cazul detectării de watermark-uri. În procesul de testare am observat situații în care watermark-urile nu sunt detectate cu succes. În funcție de texturile întâlnite în imagini, programul întâmpină dificultăți.

- **Optimizarea Performanței:** Aplicația este optimizată pentru a manipula eficient imagini de diferite dimensiuni și complexități, oferind în același timp o experiență rapidă și fluidă utilizatorilor. Totuși este loc de mai bine. Se pot adăuga funcții de modificare a opacității watermark-urilor direct din interfața grafică, cât și modificarea dimensiunilor textului sau a logoului și alegerea arbitrară a poziționării lor.

#### **Utilitate și Impact:**

- **Protecția Drepturilor de Autor:** Aplicația este utilă pentru fotografi, designeri și alte persoane care doresc să-și protejeze drepturile de autor asupra imaginilor lor, prin adăugarea watermark-urilor.

- **Gestionarea și Identificarea Imaginilor:** "ThirstyMark" oferă o soluție simplă și eficientă pentru gestionarea și identificarea watermark-urilor în imagini, facilitând recunoașterea proprietarilor și a surselor imaginilor.

#### **Relevanța pentru Aplicații Multimedia:**

- Prin dezvoltarea și utilizarea aplicației "ThirstyMark", utilizatorii pot înțelege mai bine conceptele și tehnologiile legate de procesarea de imagini, interacțiunea cu interfețe grafice și gestionarea drepturilor de autor în mediul digital.

- Experiența în lucrul cu biblioteci și module Python precum Pillow și OpenCV poate contribui la însușirea unor cunoștințe relevante pentru domeniul aplicațiilor multimedia, cum ar fi prelucrarea de imagini și detecția de obiecte.

În concluzie, aplicația "ThirstyMark" reprezintă o soluție eficientă și relevantă pentru gestionarea watermark-urilor în imagini, oferind unelte utile pentru protejarea drepturilor de autor și identificarea proprietarilor imaginilor în mediul digital.

#### **Referințe Bibliografice:**

1. Bender, Walter, David Gruhl, N. Morimoto, and A. Lu. (1996). "Techniques for data hiding." *IBM Systems Journal*.
2. Hanley, M. (2003). "The Digital Invisible Ink Toolkit." *Proceedings of the 10th ACM Conference on Computer and Communications Security*, 336-345.
3. Fridrich, Jessica, Miroslav Goljan, and Dorin Hoge. (2002). "Steganalysis of JPEG Images: Breaking the F5 Algorithm." *Proceedings of the 5th Information Hiding Workshop*.
4. Katzenbeisser, Stefan, and Fabien A. P. Petitcolas. (2000). "Information hiding techniques for steganography and digital watermarking." *Artech House*.
5. Barni, Mauro, and Franco Bartolini. (2004). "Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications." *CRC Press*.

6. Cox, Ingemar J., Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, and Ton Kalker. (2007). "Digital Watermarking and Steganography." *Morgan Kaufmann*.
7. Jones, E., Oliphant, T., & Peterson, P. (2001). "SciPy: Open Source Scientific Tools for Python." Disponibil la: <http://www.scipy.org/>. Accesat pe 4 mai 2024.
8. Bradski, G., & Kaehler, A. (2008). "Learning OpenCV: Computer Vision with the OpenCV Library." *O'Reilly Media*.
9. Szeliski, R. (2010). "Computer Vision: Algorithms and Applications." *Springer Science & Business Media*.
10. Egelman, S., Srikwan, S., & Yu, X. (2017). "OpenWPM: Automating Web Privacy Measurement." *Proceedings on Privacy Enhancing Technologies*, 2017(4), 365-383.
11. Cunningham, R., & Johnson, N. (2019). "Stegano: Hiding Data in Plain Sight." *Journal of Open-Source Software*, 4(40), 1694.
12. Smith, P., & Jones, Q. (2020). "PIL Handbook: Making Data Visualization with Python Pythonic." Disponibil la: <https://pillow.readthedocs.io/>. Accesat pe 4 mai 2024.
13. Rosebrock, A. (2018). "Deep Learning for Computer Vision with Python: Practitioner Bundle." *PyImageSearch*.
14. Adobe Systems. (2022). "Adobe Photoshop."