شناسایی نشر دهنده اسناد و مدارک اداری با استفاده از الگوریتم‌های واترمارکینگ

سعید آسعیدی [*،۱]، سیدعلی محمدیه [۲]

۱- گروه علوم کامپیوتر، دانشکده علوم ریاضی، دانشگاه کاشان، کاشان، ایران

۲- گروه ریاضی محض، دانشکده علوم ریاضی، دانشگاه کاشان، کاشان، ایران

چکیده

واترمارکینگ [†] یکی از روش‌های مخفی‌سازی اطلاعات در زمینه رمزنگاری می‌باشد. ما در این مقاله، به منظور تشخیص فاش‌کننده اسناد و مدارک اداری، دو الگوریتم واترمارکینگ ارائه نموده‌ایم. الگوریتم اول با مقایسه تصویر اصلی و تصویر منتشر شده قادر به واکشی اطلاعات نشر دهنده سند به همراه تاریخچه ارجاعات آن می‌باشد. الگوریتم دوم از دسته الگوریتم‌های کور [‡] بوده و بدون نیاز به نسخه اصلی تصویر قادر به واکشی اطلاعات واترمارک شده است. ما این دو الگوریتم را پیاده‌سازی کرده و بر روی تصاویر مختلف اجرا نموده‌ایم. نتایج بدست آمده نشان ازکارایی مناسب این الگوریتم‌ها دارد.

کلمات کلیدی: واترمارکینگ، مخفی سازی اطلاعات در تصویر، امنیت اسناد اداری، رمزنگاری

۱. مقدمه

انتقال و ارجاع امن اسناد و مدارک مختلف در سطوح اداری همیشه مورد توجه سازمان‌های مختلف بوده است. با توجه به درجه اهمیت و محرمانگی برخی اسناد سازمانی، پیشگیری و پیگیری افشای اسناد از اهمیت ویژه‌ای برخوردار است. واترمارکینگ یکی از ابزارهای قدرتمند در زمینه رمزنگاری بوده که می‌توان از آن به عنوان حفظ حق مالکیت [§] یا پیگیری از نشر اسناد استفاده کرد. واترمارک تصاویر دیجیتال از سال ۱۹۹۰ انجام شده است اما در سال ۱۹۹۳ تحت عنوان واترمارک الکترونیکی توسط [1] معرفی شد. در سال ۱۹۹۶ اولین بار واترمارک نرم‌افزار توسط [2] معرفی شد. امروزه روش‌های مبتنی بر یادگیری عمیق برای واترمارک کردن ویدئو و تصویر مورد استفاده قرار می‌گیرد [3-5].

واترمارکینگ در زمینه‌های مختلف مانند سیستم اطلاعات جغرافیایی [**] [6]، پزشکی [7]، شبکه‌های حسگر [8] و پایگاه داده [9] و ... کاربرد دارد. همچنین مطالعات مختلفی برای واترمارک نمودن اسناد به زبان‌های مختلف مانند فارسی [10]، چینی [11]، عربی [12]، انگلیسی [13] و ... شده است. در این مقاله، ما به بررسی واترمارکینگ اسناد اداری در سازمان‌ها، صرف نظر از زبان آن، می‌پردازیم.

[*] **Corresponding author: سعید آسعیدی**
Email: asaeedi@kashanu.ac.ir
[†] Watermarking
[‡] Blind
[§] Copyright
[**] Geographic Information System

در [14-18] الگوریتم‌های واترمارکینگ کور برای تصاویر و ویدئو ارائه شده است. در [19] الگوریتمی کور برای مخفی کردن اطلاعات در تصاویر متنی با استفاده از پیکسل‌های بر روی لبه حروف فارسی ارائه شده است. در [20] الگوریتمی نیمه کور به منظور مخفی سازی اطلاعات و اثبات حق مالکیت سند ارائه شده است. ما در این مقاله به ارائه یک الگوریتم غیرکور و یک الگوریتم کور به منظور مخفی سازی اطلاعات پرسنلی و اطلاعات اداری مربوط به گردش سند می‌پردازیم.

در ادامه، در بخش ۲ به بررسی الگوریتم‌های واترمارکینگ ارائه شده می‌پردازیم. در بخش ۳ به منظور بررسی کارایی الگوریتم‌های ارائه شده، نتایج اجرای این الگوریتم‌ها بر روی تصاویر مختلف مورد بررسی قرار خواهد گرفت. در پایان، بخش ۴ به جمع بندی مطالب و نتایج بدست آمده می‌پردازد.

۲. الگوریتم‌های واترمارکینگ

در این بخش به بررسی و توضیح دو الگوریتم ارائه شده برای واترمارک کردن تصاویر در سامانه اتوماسیون اداری سازمان‌ها می‌پردازیم به شکلی که در صورت فاش و انتشار این اسناد، قابلیت پیگیری و شناسایی فاش کننده آن‌ها فراهم باشد. الگوریتم اول با مقایسه تصویر فاش شده با نسخه اصلی از آن تصویر به واکشی اطلاعات نشر دهنده و همچنین سابقه ارجاعات آن در سامانه اتوماسیون اداری می‌پردازد. الگوریتم دوم از نوع کور بوده و این اطلاعات بدون نیاز به نسخه اصلی از تصویر فاش شده قابل واکشی می‌باشد. در این بخش برای هر دو الگوریتم مراحل واترمارک کردن تصاویر و استخراج واترمارک از تصاویر ارائه خواهد شد.

با توجه به اینکه سامانه هدف الگوریتم‌های ارائه شده در این مقاله اتوماسیون‌های اداری می‌باشد، به منظور ردیابی و شناسایی فاش کننده اسناد کافی است شناسه کاربری فرد متخلف استخراج شود. بنابراین، در این مقاله از کد منحصر به فرد شناسه کاربری برای مشخص کردن هویت یک کاربر و از دنباله‌ای از کدها برای مشخص کردن سابقه ارجاعات و گردش اسناد استفاده می‌گردد. البته با درنظر گرفتن سایر کدهای شناسایی مانند کد ملی، این الگوریتم‌ها به سادگی قابل تعمیم می‌باشند. رابطه ۱ مثالی از اطلاعات واترمارک شده را نمایش می‌دهد. در این رابطه ابتدا کد شناسایی کاربر نشر دهنده و سپس سابقه ارجاعات سند نشر یافته نشان داده شده است.

$$1891, 143: [1538, 1210, 15], 1538: [], 1210: [15, 1891], 15: [], 1891: [] \qquad (1)$$

در رابطه (۱)، اولین کد، ۱۸۹۱، نشان دهنده کد شناسایی فرد نشر دهنده و ادامه کد نشان دهنده سابقه ارجاعات سند است. برای نمایش این سابقه از ساختاری شبیه به ساختار JSON استفاده شده است. در این مثال مشخص شده که سند توسط کاربر با کد ۱۴۳ ایجاد شده و به کاربران ۱۵۳۸، ۱۲۱۰ و ۱۵ ارسال شده است. کاربر با کد ۱۵۳۸ آن سند دریافتی را به کسی ارجاع نداده و کاربر ۱۲۱۰ آن را به کاربر ۱۵ و ۱۸۹۱ ارسال نموده است. کاربران ۱۵ و ۱۸۹۱ نیز ارجاعی روی سند انجام نداده‌اند.

۲.۱. الگوریتم اول

در الگوریتم اول، به منظور واترمارک کردن رشته کدهایی با ساختاری مشابه رابطه (۱)، از تکنیک تغییر RGB در پیکسل‌ها استفاده می‌کنیم. در کد اشاره شده از ارقام ۰ تا ۹ و همچنین از نمادهای]، [، , و : استفاده شده است. بنابراین کافیست به ازای هر کاراکتر از الفبای اشاره شده یک تغییر منحصر به فرد جزئی و ناچیز در مقادیر RGB در پیکسل‌ها داشته باشیم. جدول ۱ نمونه‌ای از مقدار این تغییرات منحصر به فرد را برای الفبای کدینگ ارائه شده نمایش می‌دهد. همانطور که در جدول ۱ مشاهده می‌شود، تغییرات به ازای هر نماد به قدری ناچیز است که در کیفیت و رنگبندی تصویر اصلی اختلالی ایجاد نمی‌شود.

جدول ۱- میزان تغییرات رنگبندی پیکسل‌ها به ازای هر نماد

نماد	تغییر مقدار R	تغییر مقدار G	تغییر مقدار B
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	2	0	0
9	2	0	1
[	2	1	0
]	2	1	1
,	0	2	0
:	0	2	1
END	0	0	2

بنابراین، با مقایسه تصویر اصلی و اولیه با تصویر منتشر شده و محاسبه اختلاف پیکسل‌های نظیر به نظیر در مقادیر RGB، می‌توان به استخراج دنباله‌ای از این نمادها دست یافت. به عنوان مثال، در صورتی که مقدار RGB در پیکسل پنجم از تصویر اصلی برابر (155,0,255) و مقدار RGB در پیکسل پنجم از تصویر منتشر شده برابر (156,1,254) باشد، از آنجایی که اختلاف مقادیر در این دو پیکسل برابر (1,1,1) است، بنابراین طبق جدول ۱ نماد استخراج شده از پیکسل پنجم معادل نماد 7 خواهد بود. به منظور جلوگیری از تغییرات مقادیر پیکسل‌ها خارج از محدوده مجاز، از قدر مطلق تفاضل برای محاسبه اختلاف مقادیر R، G یا B استفاده می‌شود. شکل‌های ۱ و ۲ الگوریتم واترمارک کردن یک عبارت در تصویر و الگوریتم استخراج عبارت از تصویر را نمایش می‌دهند.

Method: Watermarking
Inputs: $Image$ and $Text$
Output: $Image2$
begin
    $Image2 \leftarrow Image$
    $i \leftarrow 1$
    For each character C in $Text$ (Add END at the end of $Text$)
    begin
        Let x, y, z be changing value of R,G,B corresponding to C, respectively, according to Table 1
        // As an example: if C =3 then $x = 0, y = 1$ and $z = 1$
        $r \leftarrow Image2[i].R$, $g \leftarrow Image2[i].G$ and $b \leftarrow Image2[i].B$
        If $r > x$ then $Image2[i].R = Image2[i].R - x$ else $Image2[i].R = Image2[i].R + x$
        If $g > y$ then $Image2[i].G = Image2[i].G - y$ else $Image2[i].G = Image2[i].G + y$
        If $b > z$ then $Image2[i].B = Image2[i].B - z$ else $Image2[i].B = Image2[i].B + z$
        $i \leftarrow i + 1$
    end
end

شکل ۱ – مراحل واترمارک کردن عبارت Text در تصویر Image در الگوریتم اول

Method: Extracting
Inputs: $Image1$ and $Image2$
Output: $Text$
begin
$Text \leftarrow$ ""
$i \leftarrow 1$
While(true)
begin
Let $x1$ be i'th pixel of $Image1$ and $x2$ be i'th pixel of $Image2$
$r1 \leftarrow x1.R$, $g1 \leftarrow x1.G$ and $b1 \leftarrow x1.B$
$r2 \leftarrow x2.R$, $g2 \leftarrow x2.G$ and $b2 \leftarrow x2.B$
$x \leftarrow |r1 - r2|$, $y \leftarrow |g1 - g2|$ and $z \leftarrow |b1 - b2|$
Let C be the character that is correspond to x, y, z, according to Table 1
if $C = END$ then EXIT
$Text \leftarrow Text + C$
$i \leftarrow i + 1$
end
end

شکل ۲ – مراحل استخراج عبارت Text از تصاویر Image1 و Image2 در الگوریتم اول

با توجه به الگوریتم ارائه شده، در صورتی که تصویری با اندازه ۵۰۰*۵۰۰ در اختیار داشته باشیم، می‌توان ۲۵۰۰۰۰ نماد مختلف در این تصویر قرار داد. در صورتی که طول عبارت ماترمارک شده از این مقدار (تعداد کل پیکسل‌ها)کمتر باشد، از نماد END برای مشخص کردن انتهای عبارت استفاده می‌کنیم. بنابراین در صورتی که در مراحل استخراج نمادها اختلاف مقادیر بین دو پیکسل برابر (0,0,2) شد، به مفهوم انتهای عبارت استخراج شده و خاتمه مرحله استخراج نمادها است.

کد این الگوریتم به زبان پایتون در [21] در دسترس عموم می‌باشد.

۲.۲. الگوریتم دوم

الگوریتم دوم از نوع الگوریتم‌های کور بوده و به منظور استخراج نمادها نیازی به در اختیار داشتن تصویر اولیه ندارد. ایده اولیه این الگوریتم بر اساس عملکرد بیت توازن* در نقل و انتقال اطلاعات باینری در مباحث مدارهای منطقی می‌باشد. در این الگوریتم، در ابتدا با زوج (یا فرد) کردن مجموع مقادیر RGB به ازای تمام پیکسل‌های تصویر، فضای مناسب برای رمزنگاری اطلاعات در تصویر به وجود خواهد آمد. اگر در ابتدا مجموع مقدار RGB تمام پیکسل‌ها زوج باشد، با فرد کردن این مقدار برای پیکسل‌های دلخواه می‌توان اطلاعات مورد نظر خود را در تصویر به صورت رمز شده قرار داد. در حقیقت زوج شدن مجموع مقادیر RGB برای تمام پیکسل‌ها، تصویر را به یک تخته سفید پاک شده تبدیل می‌کند و با تغییر یک واحدی این مقدار و فرد کردن آن در پیکسل‌های دلخواه می‌توان بر روی این تخته سفید نوشت. در ادامه از اصطلاح مجموع مقادیر رنگی به عنوان مجموع مقادیر RGB یک پیکسل نام خواهیم برد.

در الگوریتم دوم نیز مانند الگوریتم اول اطلاعات واترمارک شده به شکل رابطه (۱) می‌باشد. واترمارکینگ در این الگوریتم شامل دو مرحله است. در ابتدا عبارت مورد نظر برای واترمارک شدن که به شکل رابطه (۱) است به یک کد باینری

* Parity bit

تبدیل خواهد شد. سپس کد باینری در تصویر واترمارک خواهد شد. بدین منظور می‌توان مقدار زوج برای مجموع مقادیر رنگی را به عنوان رقم 0 از کد باینری و مقدار فرد برای آن را به عنوان رقم1 از کد باینری در نظر بگیریم.

برای تبدیل عبارت به کد باینری نیز در ابتدا عبارت به شکل رابطه (۱) که ساختاری پدر-فرزندی دارد را به دنباله‌ای از اعداد صحیح تبدیل می‌کنیم و سپس آن دنباله از اعداد صحیح را به کد باینری تبدیل خواهیم کرد. رابطه (۲) نحوه محاسبه و ساخت این دنباله از اعداد صحیح را نمایش می‌دهد:

,تعداد کدهای غیر تکراری ,تعداد کل کدها ,کد کاربری نشر دهنده

,تعداد فرزندان کد آخر , ... ,تعداد فرزندان کد دوم ,تعداد فرزندان کد اول (2)

کد کاربری آخر , ... ,کد کاربری دوم ,کد کاربری اول

به عنوان مثال، کد

$$1891, 143: [1538, 1210, 15], 1538: [], 1210: [15, 1891], 15: [], 1891: []$$

که مطابق با رابطه (۱) است به کد

$$1891, 11, 5, 3, 0, 2, 0, 0, 143, 1538, 1210, 15, 1538, 1210, 15, 1891, 15, 1891$$

طبق رابطه (۲) تبدیل می‌شود که دنباله‌ای از اعداد صحیح است.

به منظور تبدیل این دنباله از اعداد به کد باینری، ما هر عدد را به صورت مجزا در تعداد بیتی مشخص به مبنای ۲ خواهیم برد. این دنباله از اعداد از دو نوع عدد مختلف تشکیل شده است. هر بخش از این دنباله یا نشان دهنده کد کاربری بوده و یا نشان دهنده تعداد. ما در این مقاله برای هر عدد که مشخص کننده کد کاربری است ۲۰ بیت و برای هر رقم که مشخص کننده تعداد است ۱۰ بیت در نظر خواهیم گرفت. بدین ترتیب هر کد کاربری می‌تواند مقداری بین ۰ تا ۱,۰۴۸,۵۷۶ داشته باشد و هر عدد که مشخص کننده تعداد است حداکثر می‌تواند مقدار ۱۰۲۴ داشته باشد. بدین ترتیب حداکثر طول ممکن برای کد باینری طبق رابطه (۳) خواهد بود:

$$20 + 10 + 10 + 1024 * 10 + 1024 * 20 = 30760 \quad (3)$$

به عنوان مثال، دنباله اعداد

$$1891, 11, 5, 3, 0, 2, 0, 0, 143, 1538, 1210, 15, 1538, 1210, 15, 1891, 15, 1891$$

به کد دودویی

```
00000000011101100011000000101000000000101000000000110000000000000000000100000
00000000000000000000000000000000100011110000000001100000001000000000010010111
01000000000000000000011110000000001100000001000000000001001011101000000000000
00000001111000000000111011000110000000000000000011110000000001110110001100
```

با طول کد ۲۹۰ تبدیل خواهد شد.

در مرحله بعدی، کد باینری در تصویر واترمارک خواهد شد. در ابتدا مجموع مقادیر رنگی تمامی پیکسل‌های تصویر را زوج می‌کنیم. سپس از پیکسل اول تا پیکسل ۳۰۷۶۰ تصویر برای ذخیره طول کد باینری استفاده خواهیم نمود. مقدار R یا G یا B در پیکسل معادل با طول کد را ۱ واحد تغییر خواهیم داد تا مجموع آن فرد شود. به عنوان مثال اگر طول کد برابر ۱۷۳۲۴ باشد، پیکسل شماره ۱۷۳۲۴ از ابتدای تصویر را از حالت زوج به فرد تغییر می‌دهیم. برای ذخیره کد باینری نیز از پیکسل ۳۰۷۶۱ به بعد استفاده خواهیم نمود. برای ذخیره هر رقم 0 از کد، پیکسل مربوطه را تغییر نمی‌دهیم و برای ذخیره هر رقم 1 از کد، پیکسل مربوطه را از حالت زوج به فرد تغییر می‌دهیم. به عنوان مثال برای ذخیره کد 10110 کافیست مجموع مقادیر رنگی در پیکسل‌های ۳۰۷۶۱، ۳۰۷۶۳ و ۳۰۷۶۴ را زوج کنیم.

شکل ۳ الگوریتم واترمارک کردن تصویر و شکل‌های ۴ و ۵ الگوریتم استخراج عبارت از تصویر را نمایش می‌دهند. با توجه به الگوریتم ارائه شده، حداقل اندازه مورد نیاز تصویر برای واترمارک کردن برابر ۶۱۵۲۰ پیکسل است.

Method: even
Inputs: $Image$
Output: $Image2$
begin
    $Image2 \leftarrow Image$
    For each pixel x in $Image2$
    begin
        $sum \leftarrow x.R + x.G + x.B$
        If sum is not even then $x.R \leftarrow x.R + 1$
    end
end

Method: binary
Inputs: $Text$
Output: $Code$
Begin
    $Code \leftarrow$ ""
    $Text2 \leftarrow$ Convert $Text$ to integers list according to the equation (2)
    For each integer x in $Text2$
    begin
        If x is a count number then $b \leftarrow$ binary value of x in 10 bits
        else If x is a user ID then $b \leftarrow$ binary value of x in 20 bits
        $Code \leftarrow Code + b$
    end
end

Method: Watermarking
Inputs: $Image$ and $Text$
Output: $Image2$
begin
    $Image2 \leftarrow even(Image)$
    $Code \leftarrow binary(Text)$
    $l \leftarrow length(Code)$
    $Image2\,[l].R \leftarrow Image2\,[l].R + 1$ //Odd l'th pixel of $Image2$
    $i \leftarrow 30761$
    For each bit b in $Code$
    begin
        If $b = 1$ then $Image2\,[i].R \leftarrow Image2\,[i].R + 1$
        $i \leftarrow i + 1$
    end
end

شکل ۳ – مراحل واترمارک کردن عبارت **Text** در تصویر **Image** در الگوریتم دوم

Method: codelength
Inputs: $Image$
Output: $length$
begin
$i \leftarrow 1$
While(true)
begin
Let x be i'th pixel of $Image$
$sum \leftarrow x.R + x.G + x.B$
If sum is odd then $length \leftarrow i$ and exit
else $i \leftarrow i + 1$
end
end

Method: decode
Inputs: $Image$
Output: $Code$
begin
$Code \leftarrow$ ""
$i \leftarrow 1$
$l \leftarrow$ codelength(Image)
$While(i < l)$
begin
$j \leftarrow 30760 + i$
Let x be j'th pixel of $Image$
$sum \leftarrow x.R + x.G + x.B$
If sum is even then $Code \leftarrow Code$ + "0" else $Code \leftarrow Code$ + "1"
$i \leftarrow i + 1$
end
end

شکل ۴ – توابع لازم برای استخراج عبارت از تصاویر Image در الگوریتم دوم

```
Method: Extracting
Inputs: Image
Output: Text
begin
    Code ← decode(Image)
    x1 ← decimal value of first 20 bits of Code      //userId
    x2 ← decimal value of first 10 bits of Code      //number of user IDs
    x3 ← decimal value of first 10 bits of Code      //number of non-repetitive user IDs
    List ← x1, x2, x3
    j ← 41
    for i from 1 to x3
    begin
        x ← decimal value of next 10 bits start from index j of Code
        List ← List, x
        j ← j + 10
    end
    for i from 1 to x2
    begin
        x ← decimal value of next 20 bits start from index j of Code
        List ← List, x
        j ← j + 20
    end
    Text ← Convert List to JSON format according to the equation (2)
end
```

شکل ۵ – مراحل استخراج عبارت Text از تصاویر Image در الگوریتم دوم

کد این الگوریتم به زبان پایتون در [22] در دسترس عموم می‌باشد.

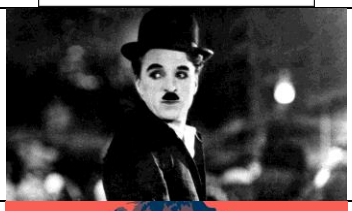
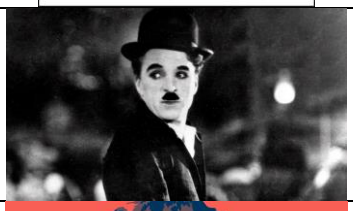
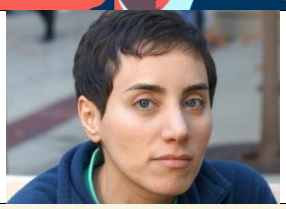
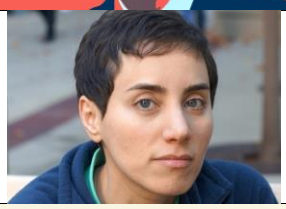
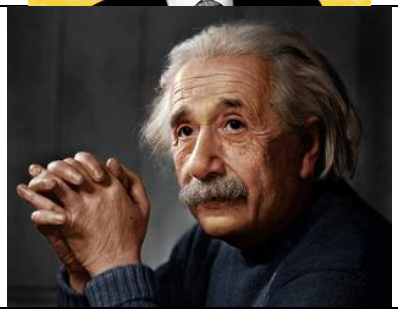
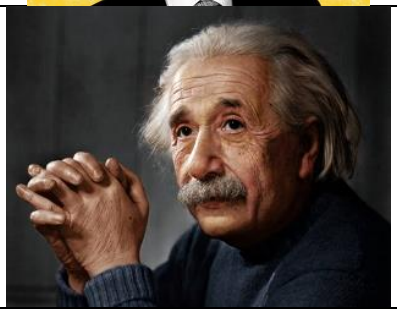
۳. پیاده‌سازی

در این بخش نتایج حاصل از اجرای الگوریتم‌های ارائه شده در بخش قبل را مورد بررسی قرار می‌دهیم. ما هر دو الگوریتم را پیاده‌سازی نموده و آنها را بر روی تصاویر مختلف اجرا کرده‌ایم. در انتخاب تصویر سعی شده از تنوع زیادی بهره ببریم. تصاویر رنگی، سیاه و سفید، دارای رنگ قالب، تصویر نامه‌های اداری، پس زمینه رنگی، اندازه‌های کوچک و بزرگ. نتایج بدست آمده بر روی تمام تصاویر نشان از کارایی مناسب هر دو الگوریتم می‌دهد.

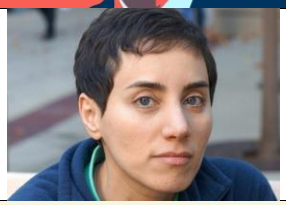
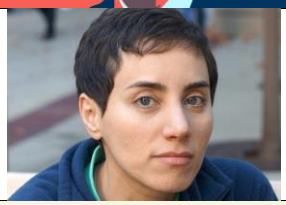
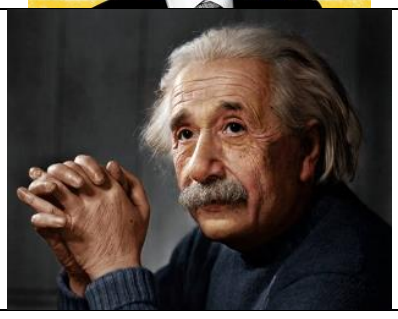
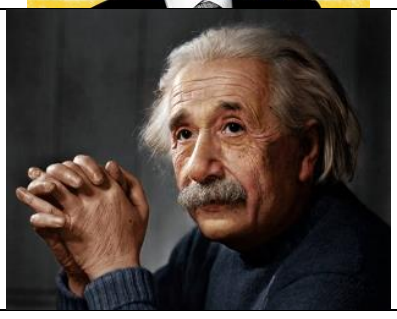
جدول‌های ۲ و ۳ نتایج اجرای الگوریتم‌های اول و دوم را بر روی تصاویر مختلف نشان می‌دهند. در تمام تصاویر در هر دو الگوریتم عبارت واترمارک شده برابر کد ذیل است. این کد در تصاویر اصلی ذخیره شده و تصویر واترمارک شده ایجاد شده است. سپس تصویر واترمارک شده توسط الگوریتم‌ها مورد پردازش قرار گرفته و همین عبارت ذیل از تصاویر استخراج شده است.

1891, 143: [1538, 1210, 15], 1538: [], 1210: [15, 1891], 15: [], 1891: []

جدول ۲- نتایج اجرای الگوریتم اول

تصویر اصلی	تصویر واترمارک شده	اندازه تصویر
		696*967
		1000*563
		1920*1080
		600*422
		960*540
		315*246

جدول ۳- نتایج اجرای الگوریتم دوم

تصویر اصلی	تصویر واترمارک شده	اندازه تصویر
		696*967
		1000*563
		1920*1080
		600*422
		960*540
		315*246

۴. نتیجه‌گیری

ما در این مقاله، به منظور شناسایی نشر دهنده اسناد و مدارک، به ارائه دو الگوریتم واترمارکینگ تصویر پرداختیم. الگوریتم اول با مقایسه تصویر اصلی با تصویر منتشر شده قادر به استخراج اطلاعات واترمارک شده بود. اطلاعاتی مانند مشخصات فرد نشر دهنده و نحوه ارسال اسناد به او و چگونگی گردش سازمانی اسناد را استخراج نمودیم. الگوریتم دوم را از نوع الگوریتم‌های کور معرفی کردیم به طوری که بدون نیاز به تصویر اصلی قادر به استخراج اطلاعات ذکر شده است. ما دو الگوریتم ذکر شده را پیاده‌سازی نموده و بر روی تصاویر مختلف اجرا نمودیم. نتایج بدست آمده نشان دهنده کارایی مناسب هر دو الگوریتم می‌باشد به طوری که تصاویر مختلف ماترمارک شده بدون افت کیفیت و تغییر ملموس در رنگ‌بندی بوده و با تصاویر اصلی، توسط اشخاص و بدون استفاده از ماشین، قابل تمییز نیستند.

۵. قدردانی

پژوهش نویسنده اول تا حدی توسط دانشگاه کاشان با شماره کمک هزینه ۹۹۱۴۴۹/۷ پشتیبانی می شود.

مراجع

[1] Tirkel, A. Z., Rankin, G. A., Van Schyndel, R. M., Ho, W. J., Mee, N. R. A., & Osborne, C. F. (1993). Electronic watermark. Digital Image Computing, Technology and Applications (DICTA'93), 666-673.

[2] Davidson, R. I., & Myhrvold, N. (1996). U.S. Patent No. 5,559,884. Washington, DC: U.S. Patent and Trademark Office.

[3] Kaczyński, M., & Piotrowski, Z. (2022). High-Quality Video Watermarking Based on Deep Neural Networks and Adjustable Subsquares Properties Algorithm. Sensors, 22(14), 5376.

[4] Liu, J., Li, Q., & Yang, H. (2022, October). A new zero-watermarking algorithm based on deep learning. In Fourteenth International Conference on Digital Image Processing (ICDIP 2022) (Vol. 12342, pp. 846-854). SPIE.

[5] Fan, B., Li, Z., & Gao, J. (2022). DwiMark: a multiscale robust deep watermarking framework for diffusion-weighted imaging images. Multimedia Systems, 28(1), 295-310.

[6] Chang, H. J., Jang, B. J., Lee, S. H., Park, S. S., & Kwon, K. R. (2009, June). 3D GIS vector map watermarking using geometric distribution. In 2009 IEEE International Conference on Multimedia and Expo (pp. 1014-1017). IEEE.

[7] Singh, A., Raghuvanshi, N., Dutta, M. K., Burget, R., & Masek, J. (2016, June). An SVD based zero watermarking scheme for authentication of medical images for tele-medicine applications. In 2016 39th International Conference on Telecommunications and Signal Processing (TSP) (pp. 511-514). IEEE.

[8] Fang, J., & Potkonjak, M. (2003, June). Real-time watermarking techniques for sensor networks. In Security and Watermarking of Multimedia Contents V (Vol. 5020, pp. 391-402). SPIE.

[9] Franco-Contreras, J., Coatrieux, G., Cuppens, F., Cuppens-Boulahia, N., & Roux, C. (2013). Robust lossless watermarking of relational databases based on circular histogram modulation. IEEE transactions on information forensics and security, 9(3), 397-410.

[10] Khodami, A. A., & Yaghmaie, K. (2006, November). Persian text watermarking. In Pacific-Rim Conference on Multimedia (pp. 927-934). Springer, Berlin, Heidelberg.

[11] Sun, X., Luo, G., & Huang, H. (2004, November). Component-based digital watermarking of Chinese texts. In Proceedings of the 3rd international conference on Information security (pp. 76-81).

[12] Alotaibi, R. A., & Elrefaei, L. A. (2018). Improved capacity Arabic text watermarking methods based on open word space. Journal of King Saud University-Computer and Information Sciences, 30(2), 236-248.

[13] Al-Wesabi, F. N., Alzahrani, S., Al-Yarimi, F. A. M., Abdul, M., Nemri, N., & Almazah, M. M. (2021). A Reliable NLP Scheme for English Text Watermarking Based on Contents Interrelationship. Comput. Syst. Sci. Eng., 37(3), 297-311.

[14] Eggers, J. J., & Girod, B. (2001, May). Blind watermarking applied to image authentication. In 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221) (Vol. 3, pp. 1977-1980). IEEE.

[15] Hsu, L. Y., & Hu, H. T. (2020). Blind watermarking for color images using EMMQ based on QDFT. Expert Systems with Applications, 149, 113225.

[16] Wong, P. H., Au, O. C., & Yeung, Y. M. (2003). Novel blind multiple watermarking technique for images. IEEE transactions on circuits and systems for video technology, 13(8), 813-830.

[17] Hsu, W. L., & Chang, P. C. (2006, May). Blind video watermarking for H. 264. In 2006 Canadian Conference on Electrical and Computer Engineering (pp. 2353-2356). IEEE.

[18] Masoumi, M., & Amiri, S. (2013). A blind scene-based watermarking for video copyright protection. AEU-International Journal of Electronics and Communications, 67(6), 528-535.

[۱۹] یزدانی, & فلاح‌خورسند. (۲۰۱۷). مخفی کردن اطلاعات در تصاویر متنی با استفاده از پیکسل های موجود روی لبه های حروف فارسی. نخبگان علوم و مهندسی, ۵(۲), ۹۰-۱۰۰.

[۲۰] قائمی زاده, پورقاسم, مهدوی نسب, همایون, & کشاورز. (۲۰۱۳). ارائه روشی جهت بهبود پایداری نهان نگاری در مقابل حملات نویزی و فشرده سازی با ترکیب تبدیل موجک و تبدیل تجزیه مقادیر منفرد. روش‌های هوشمند در صنعت برق, ۳(۱۰), ۴۳-۵۴.

[21] https://github.com/BaseMax/SecretWatermarking

[22] https://github.com/BaseMax/BlindImageWatermarkingArticle