

ない機能制限を解除されたとしても、上にある透明画像のみが保存され、外部カメラやスクリーンショットをされた時は当該画像の一部しか表示されないようにできる。図2が撮影を抑制したい画像とし、機能制限が解除された時に撮影すると図3のようになる。図4は画面をそのまま撮影した時の画像である。



図 2

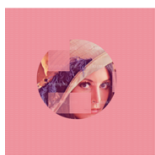


図 3

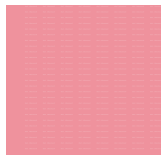


図 4

図3では円形で切り抜かれた画像の中も一部欠落している。これは当該画像を $n \times n$ に分割し、それらをランダムに点滅させているからであり、外部カメラ対策の一つである。

スクリーンショットの対策としてはcssの `sample::after` 要素で予め画面が黒くなるように設定しておき、JavaScriptの `document.body.classList.contains('sample')` を用いてcommandキー、shiftキー、windowsキーやprintscreenキーなどの要素を入力するとHTML内のbodyタグにsampleが追加されて黒くなる。

Webサイトは主にHTMLとcss、JavaScriptで構成されており、機能制限などの動的なものはJavaScriptが使用されている。そのため、ブラウザの設定からJavaScriptを無効にして表示する事で、右クリック無効の制限などは解除でき保存が可能になる。その対策が3.1節の序盤にて述べたcssで透明画像を被せる処理である。またHTML内に無効にされた場合、別のサイトに強制的に飛ばすように設計しているため当該サイトは閲覧できなくなる。

Windowsには従来のスクリーンショットの他にSnipping Toolなど外部アプリからも撮影を行うことができる。その対策として5つのキーを押さないと画面が表示されないものに加えて、マウスカーソルがサイト外に出ると自動的にモザイクが付与されることで容易には撮影できないようにしており、画面内をクリックすることで解除が可能となる。

3.2 実装結果

レンタルサーバは「ロリポップ!」[5]を利用し、ムームードメイン[6]にて独自ドメイン”<http://mosaikqr.xyz/>”を取得した。開発環境はMacBook Air 1.6GHzデュアルコアIntel Core i5を用い、使用言語はHTML、CSS、JavaScript[7]、pythonである。また、Webサイトの作成はVisual Studio Codeを用いている。

4. 評価

提案手法を実装したWebサイト(2023年4月時点)に対して15人の被験者に撮影できるかどうかの実験をしてもらった。その結果を表1にまとめる。

表1 実験端末と挑戦時間

使用端末	スマホ	Mac	Windows	合計
人数	4人	5人	6人	15人
平均時間	2分	4分	3.3分	3.1分

この際に、どの程度の画像が撮影できたのかと、行った撮影手段についても調査した。結果として、スマートフォンの4人は全てスクリーンショットで図3のような部分的撮影ができており、Macでは4人撮影を阻止できたが、1人は予め画面録画をしながらWebサイトに入ったため部分的に撮影ができた。Windowsでは2人に3.3節で述べたようにprintscreen単体で図3のように撮影され、1人がインターネット環境が悪い状態でJavaScriptを無効にすると図2のような完全な状態が撮影された。

結果としては完全な状態で撮影されたのは15人中1人であった。また、実験を通して抜け道が判明したため、対策としてHTMLにて最初に別の画像をサイト全体に重ねておき時間経過で画面から消えるように設定したため、現在では撮影を抑制できている。

5. まとめ

現代の社会は、インターネットコンテンツを誰もが自由に閲覧でき、また、容易に公開することができるため、著作権の保護が困難になっている。そこで本研究では、導入が容易、かつ安価な方法で対策できる手法を提案した。

6. 今後の展望

スクリーンショットの対策ではキーを一つ押してから判定が入るため、Macではcommand+shift+3 or 4によるスクリーンショットのため対策ができていないが、windowsではwindows+printscreenは対策が可能であっても、printscreen単体では撮影してから黒くなるため対策できていない状態である。加えて、現状ではOSのバージョンやブラウザによって対策が変わるため、今後はより強固な対策を行う必要がある。

参考文献

- [1] 記事原稿や画像の無断転載など著作権侵害の損害賠償額について解説 - 咲くやこの花法律事務所
<https://kigyobengo.com/media/useful/318.html>
- [2] パイレーツバスター AWP - ファイルや Web コンテンツの情報漏洩対策ソフト
<https://www.tsslk.jp/awp/>
- [3] NETFLIX
<https://www.netflix.com/jp/>
- [4] DRM-X.com デジタル著作権管理ソフトウェア、DRM ソフトウェア、DRM ソリューション、DRM Android、コピー防止、HTML DRM 暗号化、オーディオ/ビデオ暗号化
<https://www.drm-x.com/Default.aspx>
- [5] ロリポップ! レンタルサーバー | WordPress 利用満足度 No
<https://lolipop.jp>
- [6] ムームードメイン | 欲しいドメインがすぐ見つかる。
<https://muumuu-domain.com>
- [7] 池田泰延, 鹿野壮. JavaScript コードレシピ集. 技術評論社. 2019