

## 2 - Fonctionnalités et architectures des Benchmarks de watermark existants

### 1-Checkmark

URL :

<http://watermarking.unige.ch/Checkmark/>

Version :

1.2 (Dec. 14, 2001)

Architecture :

- Fonctionne sous Windows NT / 2000 avec une machine virtuelle java.
- Matlab 6 doit être installé.
- Le benchmark et les attaques sont codés en Matlab.
- Les fichiers XML sont traités grâce à une librairie java additionnelle.
- Le logiciel JASPER est utilisé pour la sauvegarde d'image.

Fonctionnalités :

- Le benchmark s'effectue sur une liste d'images définie par l'utilisateur.
- Checkmark calcule le temps de détection de l'empreinte ainsi que le pourcentage de réussite aux différentes attaques.
- La complexité des tests de robustesse est paramétrable.
- Il génère les résultats sous forme de fichier XML, Il peut également les mettre sous forme de tableau HTML.
- Grâce à JASPER, il sauvegarde les images attaquées au format JPEG 2000.
- Ce benchmark travaille directement sur les images préalablement watermarquées : Checkmark n'a donc pas besoin d'algorithme de tatouage.

Inconvénients :

- Pas d'interface utilisateur :
- La modification manuelle des fichiers Matlab est nécessaire pour configurer le benchmark.
- La fonction de détection du watermark doit être réécrite et implantée à la main dans Checkmark pour chaque nouvel algorithme à tester.
- Checkmark ne peut donc pas tester deux algorithmes de watermarking simultanément.
- Le portage sous Linux/Unix est possible, mais il faut apporter des modifications aux fichiers sources. Il faut également télécharger et recompiler les sources de l'utilitaire JASPER.

## 2- Stirmark

URL :

<http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/>

Version :

4.0

Architecture :

- Fonctionne sous Windows/SQL Server et Linux/Unix.
- Le code est entièrement en C++.
- Les algorithmes de watermarking doivent être passé à Stirmark sous forme de bibliothèques dynamiques (DLL pour Windows ou SO pour Linux) Les entêtes des fonctions sont normalisées.

Fonctionnalités :

- On peut créer différents profils de test pour un même algorithme de watermark. : Un fichier de configuration permet de spécifier et configurer les tests qui devront être effectués, et de choisir les images à utiliser.
- Stirmark génère des fichiers de log pour les résultats.
- La version Windows peut se connecter à SQL Server pour stocker les résultats dans une base de donnée.
- Il évalue également la qualité des images tatouées, la lisibilité de l'empreinte et les temps de traitement.

Inconvénients :

- Pas de possibilité de rajouter des attaques sans recompiler Stirmark.
- L'interface utilisateur se limite à un fichier de configuration et aux paramètres passés par la ligne de commande.
- Pas de possibilité de tester un algorithme écrit en Matlab.

### **3-Optimark**

URL :

<http://poseidon.csd.auth.gr/optimark/>

Version :

1.0

Architecture :

- Fonctionne sous Windows 95/98/2000/NT/Me/XP.
- Pour tester le watermark, on doit écrire deux programmes exécutables Windows en mode console qui doivent prendre en ligne de commande des paramètres spécifiques pour que le test s'effectue correctement.

Fonctionnalités :

- Interface Graphique (Tout est configurable grâce à de simples clics de souris)
- Evalue les performances d'un watermark avec des clés et des messages différents.
- Evalue les performances de détections de l'empreinte.
- Trace les courbes caractéristiques des traitements.
- Calcul les probabilités de détecter l'empreinte.
- Evalue la qualité de l'image tatouée.
- Les résultats sont mis à disposition sous forme de site WEB ou par des graphiques en JPEG ou encore, par du texte brut.

Inconvénients :

- Pas de possibilité de rajouter ses attaques.
- Pas de possibilité de tester un algorithme écrit en Matlab.
- Pas de portage sous Unix/Linux.
- Optimark n'est pas Open Source.

## 4- Certimark

URL :

<http://vision.unige.ch/certimark/>

Version :

Inconnue.

Architecture :

Inconnue.

Fonctionnalités :

- Evalue le watermark de façon flexible : en effet, on peut spécifier les paramètres de chaque watermark de façon illimitée (la fonction de marquage ainsi que celle de détection) que ce soit la taille maximum des messages, les clés...
- Distingue les différents types de watermark et effectue des tests en fonctions (watermarking publiques privés, 'blind' non 'blind', lisibles détectables et supprimables)
- Effectue les tests de robustesse standard sur la modification de l'image, ainsi que les tests de robustesse sur les attaques intensionnelles (suppressions, désactivations de détection, cryptographiques et sur le protocole)
- Effectue les tests de qualité standard par rapport à l'image d'origine (DSCQS, PSNR...) trace les courbes ROC.
- Effectue des tests de probabilité BER (Bit Error Rate)

Inconvénients :

Programme pas encore disponible.