DOI: 10.5769/C2013008 or http://dx.doi.org/10.5769/C2013008

Investigação de Crimes Relacionados à Pedofilia Utilizando Metadados de Imagens

João M. Ceron, Paulo César Herrmann Wanner, Lisandro Z. Granville, Bruno Werneck

Resumo — A investigação de crimes relacionados à pedofilia é um processo complexo que compreende a análise de imagens. Nota-se, no entanto, que cada vez mais dispositivos móveis são utilizados para a captura de imagens. Boa parte dos celulares e *tablets* possuem sistema de GPS interno o qual permite que fotos sejam capturadas com informações de localização. Valendo-se disso, este trabalho busca apresentar uma arquitetura para coleta, análise e correlação dos dados analisando informações embutidas nas fotos. Para isso, os autores do trabalho descrevem uma arquitetura e implementam um protótipo para auxiliar peritos na resolução de crimes relacionados com pedofilia.

Keywords—exif; metadada; investigação; pedofilia;

I. Introducão

A violações de leis criminais que envolvem o conhecimento de tecnologia para a sua perpetração está em franco crescimento. Dispositivos informatizados são frequentemente utilizados em alguma etapa das diferentes tipificações criminais [1].

Em crimes relacionados a pedofilia, em especial, a análise de evidências digitais pode ressaltar características fundamentais para a resolução de um crime. A análise de fotos existente num telefone celular podem, por exemplo, revelar dados do autor e até mesmo informações geográficas.

Analisar um dispositivo informático com imagens relacionadas à pedofilia é uma tarefa árdua e psicologicamente desgastante. Nesse processo, muitas vezes o investigador passa por inúmeros diretórios de fotos com conteúdo sexual envolvendo menores. A utilização de ferramentas para auxiliar nesse processo de investigação, de fato, pode facilitar a perícia criminal.

Essa demanda pode ser preenchida por ferramentas que classificam imagens segundo características intrínsecas. Existem ferramentas especializadas em reconhecer padrões de imagens, e outras em mapear características inseridas no próprio arquivo da imagem. No entanto, as atuais ferramentas apresentam lacunas que podem ser endereçadas para aprimorar investigações digital.

Sendo assim, este trabalho descreve uma arquitetura especializada na análise de metadados de imagens com o objetivo de auxiliar na investigação de crimes relacionados a pedofilia. Para isso, os autores descrevem uma arquitetura baseada no modelo cliente servidor que possibilita a coleta de informações e o armazenamento dos dados numa base centralizada. Deseja-se, deste modo, possibilitar que informações de diferentes investigações possam ser armazenadas e correlacionadas entre si.

Toda a arquitetura baseia-se apenas nos dados presentes no metadados das imagens periciadas. As informações do metadados são analisadas e exportadas para uma base de dados, como por exemplo: data da captura da imagem; configurações da câmera; e coordenadas geográficas. Tais informações das imagens agrupadas com demais dados da investigação são armazenadas numa base de dados. Dessa forma, por meio de uma interface (API), um investigador pode consultar os dados presentes na base de dados e correlacionar as informações. Adicionalmente, diferentes meios de visualização podem auxiliar na resolução da investigação:

a) *Mapas:* localização onde as fotos foram tiradas ilustrando a área de atuação do autor (ex. *Google Maps*, *Bing Maps*);
b) *Estabelecimento*: locais próximos que podem auxiliar na investigação, por exemplo, solicitar imagens de câmeras de vigilância de estabelecimentos comerciais (ex. *Google Places*, *Foursquare*);
c) *Redes Sociais:* comentários próximos ao local das fotos podem revelar observações de transeuntes (*Twitter*, comentários georreferenciados);
d) *Dispositivos:* classificação de fotos por dispositivo o que permite identificar as câmeras utilizadas.

Como resultado, os autores deste trabalho apresentam um protótipo da solução e descrevem o seu uso numa investigação de crime relacionado a pedofilia. Este trabalho está organizado da seguinte maneira: na seção II são descritos os trabalhos relacionados; na seção III é apresentado a solução proposta; na seção IV é descrito o protótipo implementado e um estudo de caso; por fim, as conclusões são apresentadas na seção V.

II. Ferramentas de Investigação

Durante uma investigação, o perito criminal precisa trabalhar com uma grande quantidade de dados: arquivos de texto, imagens, vídeos, programas executáveis, base de dados e outros. Torna-se um grande desafio extrair dessa massa de dados informações que podem comprovar uma atividade criminosa.

A fim de identificar informações relevantes para uma investigação são utilizadas ferramentas capazes de analisar diferentes tipos de arquivos em diferentes sistemas de armazenamento. Existem ferramentas que são amplamente

utilizadas para processar uma grande quantidade de dados categorizando e indexando informações encontradas a fim de facilitar o trabalho pericial. A diversidade de ferramentas para análise forense é muito alta. Boa parte dessas ferramentas são comerciais, mas também existem soluções gratuitas de boa qualidade [2].

A comunidade acadêmica constantemente busca identificar novas tendências e avalia a aplicabilidade de certas ferramentas no contexto de uma investigação. Por exemplo, em [3] os autores avaliam a ferramenta *Forensic ToolKit* (FTK) muito popular entre os peritos. A FTK é uma suíte de ferramentas para analisar um sistema de arquivos, incluindo a identificação de arquivos removidos e diversos relatórios com característica dos dispositivos. A ferramenta *Encase*, talvez a mais conhecida entre os peritos, é estuda por Garber em [4]. Assim como o FTK, a ferramenta Encase também é uma suíte de ferramentas para investigação forense fortemente customizável apresentando, até mesmo, uma linguagem de programação *script* para automatização de tarefas.

Outra gama de ferramentas forenses são as especializadas em buscam extrair informações de arquivos específicos. Tais ferramentas visam interpretar dados de arquivos de registro do Windows; informações de programas de comunicação instantânea; registro de eventos do sistema operacional; e ainda recuperar arquivos do espaço não alocado no disco utilizando técnicas de *data carving* [5]. Num contexto mais específico, algumas ferramentas são capazes de identificar imagens com pornografia infantojuvenil. O *NuDetective Forensic Tool* [6], por exemplo, é uma ferramenta especializada na busca de imagens com nudez infantil. O *NuDetective* utiliza técnicas de busca por arquivos conhecidos com base em listas de resumos criptográficos (*Hash List*), técnica também utilizada por ferramentas comerciais de amplo uso.

Observa-se, também, ferramentas com nicho específico de atuação. A ferramenta *Cellebrite Universal Forensic Extraction Device (UFED)* [7] é um exemplo. Essa ferramenta é especializada na análise de dispositivos móveis como celulares e *tablets*. Tal ferramenta faz uma varredura no aparelho e disponibiliza diferentes tipos de relatórios. Além de identificar fotos capturadas, a ferramenta disponibiliza uma mapa com o histórico de localização do dispositivo (baseado nas coordenadas GPS e torres de celular).

Apesar da flexibilidade das ferramentas forenses as mesmas apresentam certas lacunas que, em escopos específicos, ficam evidenciadas. Numa investigação cujo objetivo é analisar metadados das imagens de uma maneira mais detalhada as ferramentas generalistas de investigação forense apresentam limitações. Por exemplo, é possível obter metadados de imagens na grande maioria de ferramentas e diferentes sistemas operacionais (vide Fig. 1) no entanto as mesmas não permitem, de forma direta, correlacionar dados ou, até mesmo, atuar nos dados embutidos nas imagens.

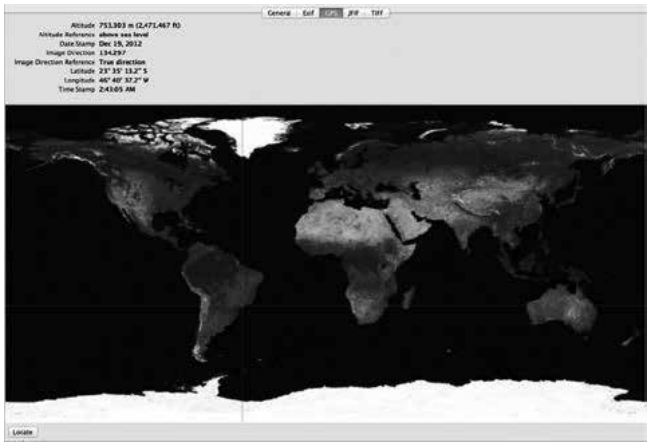

Fig. 1. Metadados de uma foto visualizada na interface nativa do sistema operacional MAC OS X v10.8 Mountain Lion.

Diante do exposto, cabem aprimoramentos nas atuais soluções para as necessidades reais definidas no escopo deste trabalho. Desenvolver uma arquitetura que possa ser utilizada como base de dados para diferentes investigações cujo objetivo seja investigar crimes relacionados a pedofilia continua sendo uma lacuna a ser explorada. Na sequência é descrita a arquitetura proposta pelos autores a fim de endereçar as atuais limitações das ferramentas existentes.

III. Arquitetura da Ferramenta de Investigação

Uma investigação forense pode atuar em diferentes tipos de mídias e formatos de arquivos de imagens. Da mesma forma, sabe-se que o universo periciável é bastante dinâmico e sofre constantes mudanças com a inserção de novas tecnologias. Sendo assim, a arquitetura proposta foi definida utilizando o conceito modular a fim de ser facilmente extensiva para adequar-se as diferentes necessidades de investigação.

Além da modularidade, é importante que a arquitetura seja concebida para adequar-se a diferentes domínios administrativos. Por exemplo, na Fig. 2 é ilustrado um cenário com diferentes domínios administrativos representados por DEICC -- (Delegacia Especializada de Investigações de Crimes Cibernéticos) -- de 3 regionais RS, DF, SP. Os diferentes domínios administrativos podem atuar segundo o conceito de federação, e também podem atuar de forma independente. Isso significa que, no exemplo, uma delegacia (DEICC) pode atuar de forma isolada, dependendo apenas do modo de operação da arquitetura configurado.

As federações representadas na Fig. 2 comunicam-se entre si via comunicação segura (VPN). Além Além disso, é descrito um módulo opcional de gerenciamento da arquitetura. Esse módulo gerencial só faz sentido quando são utilizadas diferentes federações onde deseja-se configurar a interação entre as mesmas.

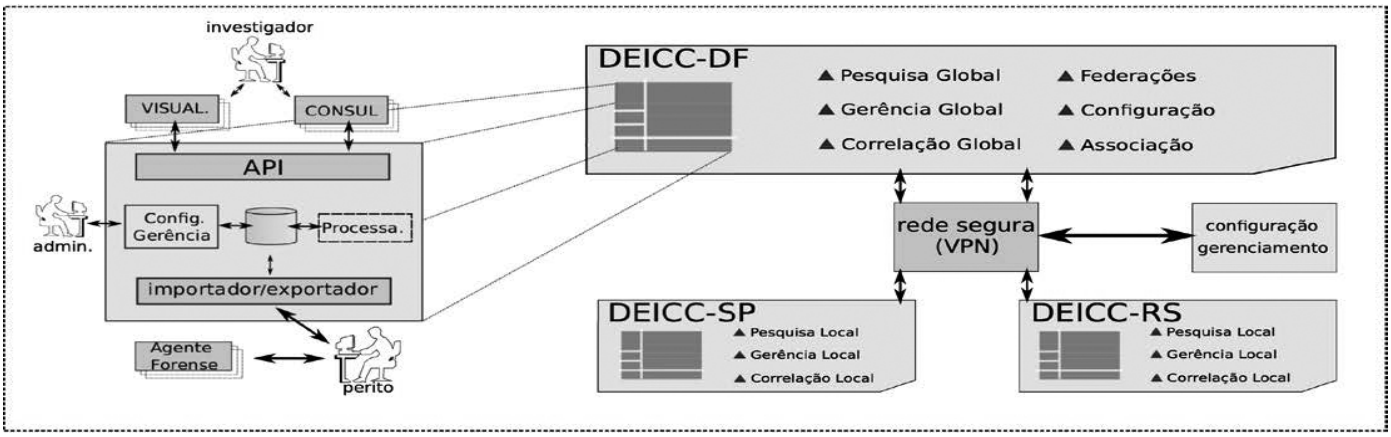

Fig. 2. Arquitetura proposta utilizando um cenário com diferentes docmínios administrativos.

Com isso, o usuário Administrador da arquitetura pode gerenciar acessos, monitorar, e configurar os diferentes módulos. Assim como o usuário Administrador, são definidos outros perfis de usuários conforme descrito abaixo:

a) *Administrador:* responsável por definir controles de acesso a base de dados e configuração. Dessa forma, torna-se possível restringir o acesso aos dados tendo em vista as permissões de um perito.

b) *Investigador*: neste perfil, o usuário pode consultar a base de dados e correlacionar informações distribuídas pela arquitetura. Por exemplo, pode-se verificar a existência de um *hash* criptográfico em diferentes investigações.

c) *Perito:* este usuário é responsável pela coleta de evidência no dispositivo periciado. O perito executa o software e faz a submissão dos dados para a base de dados.

Os diferentes módulos que compõe a arquitetura podem ser observados em destaque na Fig. 2. Na parte superior são especificados módulos para consulta e visualização dos dados armazenados na base de dados; na parte central os elementos responsáveis pelo processamento das informações e configurações da base de informações; e, por fim, na parte inferior os agentes forenses responsáveis pela coleta de dados nos dispositivos periciados.

A visualização de dados (VISUALI.) e consulta de informações (CONSUL.) permitem que o usuário com perfil Investigador consulte os dados presentes na arquitetura e faça correlações. Para que diferentes módulos de visualização e consulta pudessem ser implementados, foi desenvolvida uma interface de programação (API).

Utilizando a API desenvolvida é possível realizar consultas via linha de comando (CLI) e também via interface Web dando maior flexibilidade a arquitetura. Um exemplo de uso da API é apresentado na seção IV, no entanto a descrição da API foge do escopo desse trabalho e será endereçada em trabalhos futuros.

Os módulos responsáveis pelo controle e processamento das informações estão alinhados na parte central da arquitetura sendo respectivamente: API de consulta; importação/exportação; configuração e controle. Utilizada como interface entre os módulos de visualização e consulta, a API de consulta tem acesso a base de dados observando as políticas de controle definidos no módulo *configuração e controle*. O controle de acesso descreve políticas de acessos ao banco de dados. Essas políticas podem ser definidas segundo usuários ou valendo-se do conceito de *visões*. As visões devem considerar quesitos como permissão do usuário, operações de investigações, domínios administrativos (federações) e outros. Além dos controles, demais configurações são definidas por esse módulo, tais como: usuários e configuração da arquitetura.

O módulo importação/exportação é responsável obter as informações dos agentes forenses e armazená-las na base de dados. De forma complementar, o módulo possibilita que a base de dados seja exportada para uma outra base de dados remota, quando a arquitetura for configurada de forma hierárquica -- federações.

O Agente Forense, por sua vez, é o software responsável analisar os metadados de imagens e extraí-las para uma base de dados. Para isso, o software examina o sistema de arquivos de um dispositivo e identifica metadados no formato *Exif* (*Exchangeable Image File Format*) [8].

O *Exif* é uma especificação que define um conjunto de dados que podem ser embutidos em arquivos de imagem e áudio. Essa especificação é bastante abrangente e atualmente é implementada por diversos dispositivos, como celulares, câmeras digitais, *tablets* e outros.

Além de informações técnicas de uma imagem, o *Exif* pode opcionalmente conter informações georreferenciadas (utilizando o GPS do dispositivo). A Listagem 1 ilustra parcialmente o metadados de uma fotografia capturada utilizando uma telefone celular.

```
ExifTool Version Number : 8.90
File Name : iphone5.png
MIME Type : image/jpeg
Make : Apple
Camera Model Name : iPhone 5
Exposure Time : 1/15
F Number : 2.4
ISO : 800
Exif Version : 0221
Date/Time Original : 2013:01:14 23:41:48
Create Date : 2013:01:14 23:41:48
GPS Altitude : 80.1 m Above Sea Level
GPS Latitude : 41 deg 1' 52.80" N
GPS Longitude : 73 deg 46' 27.60" W
Image Size : 3264x2448
```

Listagem 1: Conjunto parcial de informações embutidas numa imagem descrita pela especificação *Exif*.

As informações contidas nas imagens, tais como, dispositivo utilizado, data da foto e localização podem ser fundamentais numa investigação. Compondo esses dados, um perito pode identificar os dispositivos utilizados bem como a região geográfica onde o possível crime ocorreu. Sabe-se que informações do metadados podem ser adulteradas, no entanto, essa análise está fora do escopo do nosso trabalho [9].

Numa primeira análise, o *Agente Forense* identifica imagens com informações georreferenciadas. Na sequência, o metadados dessas imagens é inspecionado bem como outras informações do arquivo são extraídas (nome do arquivo, assinatura *hash* do arquivo, e outras). Como resultado, o *Agente Forense* é responsável por compilar as informações encontradas e armazená-las num arquivo de texto num formato aberto e padronizado para exportação (XML, JSON). Da mesma forma, são incorporadas ao arquivo de exportação informações sobre a investigação e identificação do perito. Por fim, o perito tem um conjunto descritivo das informações encontradas no dispositivo e pode exportar diretamente para a base de dados centralizada.

É importante destacar que toda a arquitetura atua apenas com a análise de metadados dos arquivos. Os arquivos originais não são alterados tampouco transferidos para arquitetura. Na sequência os demais elementos da arquitetura são descritos com maior detalhamento.

IV. Implementação e Estudo de Caso

Baseando-se na arquitetura apresentada na seção anterior foi implementado um protótipo do sistema. Esta seção descreve detalhes de implementação bem como um estudo de caso com a aplicação do protótipo. Por questões de organização esta seção está organizada da seguinte forma: na primeira etapa é descrita a implementação do protótipo que foi dividida em *Módulos de Coleta* e *Módulos de Processamento*; e na segunda etapa um estudo de caso é apresentado.

A. Módulos de Coleta

Os módulos de coleta são aqui representados pelos *Agentes Forense*. O *Agente Forense* é o software responsável por varrer um sistema de arquivo em busca de metadados de imagens no formato *Exif*.

No estágio inicial da varredura o software solicita de forma interativa algumas informações para o perito, tais como: identificação do investigador, identificação da investigação, e identificação da prova a ser periciada. Com base a isso, o software produz um arquivo texto no formato aberto -- JSON ou XML -- com todas as informações inseridas pelo perito e também com os metadados encontrados na varredura. Um fragmento do arquivo gerado é ilustrado na Listagem 2.

É importante que o *Agente Forense* seja portável e com baixo nível de complexidade para ser executado nos mais diferentes cenários de investigação. Para isso, o mesmo foi implementando utilizando a linguagem *Perl* com bibliotecas específicas para lidar com informações de metadados de imagem [10].

```
<item>
  <created:time>2013-06-02 01:00:09 +0000</created:time>
  <operation>Chakal_2013</operation>
  <researcher>SP_BR_0324A</researcher>
  <md5>d07625f10c9c85c70d97880dfe81c713</md5>
  <filename>iPhone5.png</filename>
  <exif:captured>2013:05:04 18:58:13</exif:captured>
  <exif:geo:lat>-23.62754</exif:geo:lat>
  <exif:geo:long>-46.66082</exif:geo:long>
  <exif:make>Apple</exif:make>
  <exif:model>iPhone 5</exif:model>
  ....
  ...
<item>
```

Listagem 2: Fragmento do arquivo XML gerado pelo *Agente Forense*.

Na Listagem 2 é possível identificar alguns campos armazenados do arquivo resultante de uma varredura do *Agente Forense*. Observa-se dados que identificam a investigação (*operation*) e o perito (*researcher*). Adicionalmente, observa-se informações sobre o metadados da imagem, como data da captura, fabricante do dispositivo, informações geor-referenciadas, entre outras.

De posse do arquivo gerado pelo *Agente Forense*, cabe ao perito submeter o arquivo para os demais módulos de processamento de dados.

B. Módulos de Processamento

Os módulos de processamento correspondem a parte mais complexa do sistema, onde os dados coletados pelos agentes são armazenados e posteriormente processados para investigação e correlação.

Toda a implementação do protótipo foi realizada num único servidor. O servidor consiste num máquina Linux com recursos modestos, utilizando uma base de dados Sqlite versão 2.8.17. Os *Módulos de Processamento* foram implementados na linguagem *Perl* acessando a base de dados via DBI (*The Perl Database Interface Module*). Diferentemente dos demais módulos, o módulo de visualização implementado (mapa)valeu-se de *JavaScript* executado localmente no servidor Web Nginx 14.0.

De forma inicial, os *Módulos de Processamento* analisam o arquivo submetido pelo perito e armazena as informações no banco de dados da arquitetura. As informações do arquivo JSON/XML são convertidas em diferentes campos do banco de dado, dando maior flexibilidade para acesso dos dados. A interface de consulta via linha de comando, por exemplo, dá a possibilidade para um investigador consultar informações utilizando filtros. A Listagem 3 exibe uma consulta na base de dados solicitando a todos os dispositivos presentes.

```
# list_devices
 id | device                | insert_timestamp     | operation_tag    |
----+-----------------------+----------------------+------------------ +
001 | Apple, iPhone 5       | 2012:01:04 18:56:57  | Chakal_2013      |
002 | Asus, Nexus 7         | 2012-10-24 22:46:55  | Chakal_2013      |
003 | Motorola, MB525       | 2012-10-24 22:46:55  | Chakal_2013      |
004 | Apple, iPhone 5       | 2013-04-14 08:16:11  | Chakal_2013      |
```

Listagem 3: Listagem de dispositivos presentes na base de dados.

É possível observar na Listagem 3 os diferentes dispositivos presentes na base de dados e o horário de inserção no sistema, bem como, a identificação do operação. De forma direta, essas informações representam que foram analisados diferentes dispositivos na investigação *Chakal_2013* e foram encontradas fotos georreferenciadas em 4 diferentes dispositivos.

Diferentes filtros podem ser aplicados diretamente no metadados das imagens. Assim como consultar a base de dados por um dispositivo específico, pode-se buscar por *hash* criptográfico de fotos, data de captura da foto, e região geográfica. Na Listagem 4, por exemplo, são representadas duas diferentes consultas: a primeira procura todos os dispositivos do tipo ``Apple, iPhone 5'' inseridos na base de dados no último ano. E, a segunda consulta, lista todas as informações das fotos extraídas no dispositivo especificado. O comando *list_photos* exibe informações das fotos cuja localidade seja o estado de SP (segundo API de localização utilizando Google Places [11]) e que tenham sido capturadas no ultimo mês.

```
# list_photos -device="Apple,iPhone 5" --last-year
 id | device | photos | operation_tag |
-----+--------------------+---------+--------------------+
001 | Apple,iPhone 5 | 15 | Chakal_2013 |
004 | Apple,iPhone 5 | 34 | Vortice_2013 |
# list_photos -device-id="001" --local="SP" --last-month
MD5 | capture_timestamp | lat | long |
----------------+-------------------------+-------------+------------+
d076....c713 | 2013-05-04 18:58:13 | -23.62754 | -46.66082 |
a3d1....0528 | 2013-05-19 16:17:04 | -23.61047 | -46.66833 |
8a8f....c212 | 2013-05-07 10:23:50 | -23.60339 | -46.68374 |
bcac....d107 | 2013-05-29 10:23:51 | -23.60839 | -46.69112 |
```

Listagem 4: Uso da interface via linha de comando para consultar informações armazenadas na base de dados.

De forma complementar a interface de linha de comando, foi desenvolvida a interface via navegador Web. Essa interface é importante, pois permite que os dados presentes na base de dados possa ser correlacionados com outros serviços Web descrevendo o conceito de *mashups* [12].

A Fig. 3 representa uma configuração da interface de visualização implementada. No mapa da figura são ilustrados pontos representando a localização de fotos capturadas por um dispositivos periciado. É possível observar que a localização das fotos concentram-se no estado do Rio Grande do Sul e São Paulo. Logo, as fotos presente no dispositivo podem identificar características de um crime que aconteceu nos dois estados.

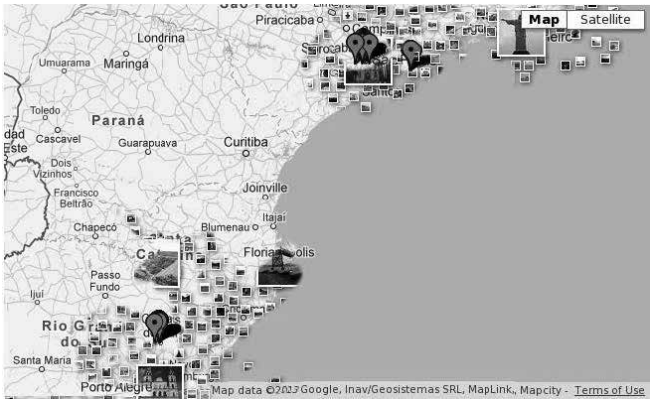

Fig. 3. Interface ilustrando os pontos onde as fotos foram capturadas e composta com uma camada de fotos do Google Panoramio.

Na sequência o protótipo implementado, descrito nesta seção, é utilizado numa investigação simulada demonstrando possíveis usos da ferramenta.

C. Estudo de Caso

Esta subseção visa ilustrar as principais funcionalidades da arquitetura. Para isso foram utilizados dados sintéticos de 3 dispositivos móveis distintos. Dessa forma, nenhuma imagem com pornografia infantil foi analisada, mas sim imagens arbitrárias georreferenciadas armazenadas nos dispositivos. Os seguintes dispositivos foram analisadas:

- Motorola Defy: celular com 40 fotos georreferenciadas dispersas no estado de São Paulo e Rio Grande do Sul;
- Asus Nexus 7: *tablet* com 10 fotos georreferenciadas no estado de São Paulo;
- iPhone 5: celular com 15 fotos dispersas no estado do Rio Grande do Sul.

Em nosso estudo, os diferentes dispositivos foram montados numa máquina e o *Agente Forense* implementado foi executado.

Inicialmente, o *Agente Forense* fez uma varredura completa no dispositivo e salvou os metadados no formato XML, conforme exemplificado na Listagem 2. Sendo assim, foi criado um arquivo com informações dos metadados para cada dispositivo: 2013-02-28_12-24-01_503_apple_dev.xml, 2013-02-28_12-30-34_138_motorola_dev.xml, 2013-02-28_12-42-11_107_asus_dev.xml.

Os arquivos XML com informações dos metadados encontrados foram submetidos para o servidor onde os módulos de processamento foram implementados. A submissão dos arquivos foi feita utilizando o SSH2, no entanto

outros meios poderiam ser utilizados, como uma interface Web de submissão. Com isso, o módulo de importação dos dados faz uma análise no arquivo XML e o armazena na base de dados. A partir desse momento, as informações coletadas pelo perito podem ser consultadas via diferentes interfaces implementadas.

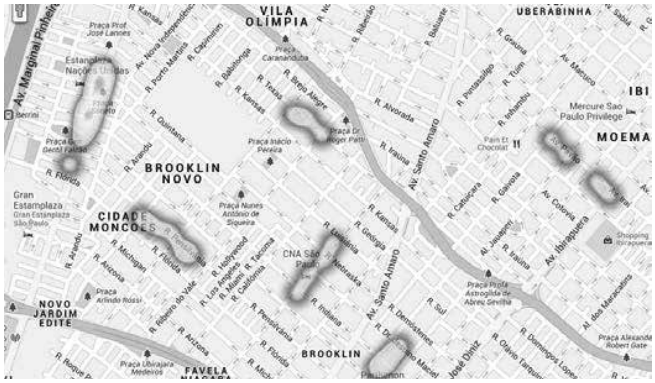

Fig. 4. Mapa utilizando gradiente de cores (*heatmap*) representando a concentração geográfica onde as fotos analisadas foram capturadas.

Na arquitetura proposta, as consultas são realizadas utilizando uma API de consulta (vide seção III). Com a mesma API, diferentes módulos de visualização podem acessar a base de dados da arquitetura. Na Listagem 3, por exemplo, foi exemplificado uma consulta utilizando a interface de comando (CLI). No entanto, outras formas de visualização podem ser implementadas. Em nosso protótipo foi desenvolvida uma interface de mapas para filtrar e ilustrar a localização das fotos. Para isso, as coordenadas geográficas das fotos -- armazenadas na base de dados – são processadas e convertidas num arquivo GeoRSS [13] e posteriormente exportadas para a interface do aplicativo Google Maps [11].

Na interface de mapas implementada é possível exprimir diferentes representações. Uma representação delas ilustra a visão geral de onde as fotos foram capturadas (vide Fig. 3). Já em outra configuração, um mapa com outras características dos dados pode ser detalhado. Na Fig. 4, por exemplo, é ilustrado um mapa no formato *heatmap*. Neste formato, é utilizado um gradiente de cores para representar a concentração geográfica onde as fotos foram capturadas. Essa informação pode identificar local por onde o suspeito passou e, até mesmo, auxiliar na identificação de um possível logadouro onde o crime foi efetivado.

Assim como a interface de linha de comando, a interface de mapas implementada permite o uso de filtros. Logo, torna-se possível plotar informação de um dispositivo específico, uma operação em particular, ou até mesmo, filtrar pela data de captura das fotos.

Uma outra representação da interface de mapas é ilustrada na Fig. 5. Nesta figura é possível definir pontos de interesse próximo ao local de captura de fotos. Através de uma pré-configuração na interface Web, o investigador pode procurar por estabelecimentos específicos para adicionar a plotagem. No exemplo, são ilustrados pontos que correspondem a escolas.

A representação de estabelecimentos próximos ao local das fotos capturadas pode auxiliar na resolução de crimes. Tem-se o conhecimento onde investigações foram solucionadas com ajuda de estabelecimentos próximos. Em uma dada investigação um perito encontrou fotos de pedofilia num dos dispositivos analisados. Nas fotos eram expostas cenas com crianças utilizando um uniforme escolar específico. Utilizando a arquitetura proposta, seria possível identificar a área de atuação de um criminoso e também escolas próximas que poderiam auxiliar na investigação.

Fig. 5. Mapa utilizando gradiente de cores (*heatmap*) representando a concentração geográfica onde as fotos analisadas foram capturadas.

O protótipo implementado possibilitou demonstrar as principais características da arquitetura proposta. São ressaltados alguns recursos que podem contribuir numa investigação. Em especial, destaca-se como fontes externas de informações georreferenciadas podem ser compostas aos metadados das fotos numa investigação criminal.

V. Considerações Finais

Nos últimos anos, no entanto, o campo de fotografia digital foi impulsionado com o uso de dispositivos móveis. Como diferencial, os dispositivo móveis podem embutir informações geográficas nas próprias fotos. Logo, analisar os metadados de fotos capturadas com dispositivos móveis tornou-se atrativo.

Embora algumas ferramentas permitam a visualização dos dados de localização, não se tem conhecimento de uma ferramenta específica para tal propósito. Sendo assim, este trabalho apresentou uma arquitetura baseada no metadados de imagens georreferenciadas para a investigação de crimes relacionados à pedofilia.

Como resultado, os autores implementaram um protótipo da arquitetura e demonstraram como informações de fotos georreferenciadas podem auxilar na resolução de casos. A interface implementada permitiu que as informações geográficas de fotos periciadas pudessem ser compostas com outros serviços *online* (*mashup*). Com isso, um investigador pode visualmente identificar características do possível crime.

Adicionalmente, acredita-se com um maior número de investigações armazenadas na base de dados seja possível

usar a própria arquitetura como base de conhecimento. Em investigações futuras, crimes semelhantes podem ser correlacionados e características reincidentes podem ser mapeadas de forma direta.

Por questões de *design*, a arquitetura proposta atua somente em imagens com informações geográficas embutidas. Como limitações, as funcionalidades descritas não são efetivas em imagens não georreferenciadas.

Em trabalhos futuros, os autores desejam implementar outras interfaces de visualização e permitir a correlação de dados com outras fontes de informação.

Referências

[1] E. Wendt e H. V. N. Jorge, Crimes Cibernéticos Ameacas e procedimentos de investigacao. Brasport, 2012.

[2] B. Carrier, "The sleuth kit.". Disponível em: http://www.sleuthkit.org/sleuthkit/desc.php, 2013.

[3] X. Ding e H. Zou, "Time based data forensic and cross-reference analysis," in Proceedings of the 2011 ACM Symposium on AppliedComputing. ACM, 2011, pp. 185–190.

[4] L. Garber, "Encase: A case study in computer-forensic technology," IEEE Computer Magazine January, 2001.

[5] M. I. Cohen, "Advanced jpeg carving," in Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, p. 16.

[6] M. de Castro Polastro e P. da Silva Eleuterio, "Nudetective: A forensic tool to help combat child pornography through automatic nudity detection," in Database and Expert Systems Applications (DEXA), 2010 Workshop on, 2010, pp. 349–353.

[7] R. Mislan, "Cellphone crime solvers," Spectrum, IEEE, vol. 47, no. 7, pp. 34–39, 2010.

[8] Technical Standardization Committee on AV & IT Storage Systems and Equipment, "Exchangeable image file format for digital still cameras: Exif Version 2.2," Tech. Rep. JEITA CP-3451, April 2002.

[9] T. Gloe, "Forensic analysis of ordered data structures on the example of jpeg files," in Information Forensics and Security (WIFS), 2012 IEEE International Workshop on. IEEE, 2012, pp. 139–144.

[10] Phil Harvey, "Image::ExifTool: Módulo para processamento de metadados de arquivos de mídia". Disponível em: http://www.sno.phy.queensu.ca/~phil/exiftool/, 2013.

[11] Google, "Google Maps API". Disponível em: http://code.google.com/apis/maps/.

[12] C. Santos, R. Bezerra, J. Ceron, L. Granville, e L. Rockenbach Tarouco, "On using mashups for composing network management applications," Communications Magazine, IEEE, vol. 48, no. 12, pp. 112–122, 2010.

[13] Y.-F. R. Chen, G. Di Fabbrizio, D. Gibbon, S. Jora, B. Renger, e B. Wei, "Geotracker: geospatial and temporal rss navigation," in Proceedings of the 16th international conference on World Wide Web, ser. WWW '07. New York, NY, USA: ACM, 2007, pp. 41–50. Disponível em: http://doi.acm.org/10.1145/1242572.1242579.