Multimed Tools Appl (2014) 68:95–110
DOI 10.1007/s11042-012-1164-3

INACT—INDECT Advanced Image Cataloguing Tool

Michał Grega · Damian Bryk · Maciej Napora

Published online: 7 July 2012

Abstract Child pornography possession and distribution are crimes which are prosecuted in most countries around the world. In some cases the law is so strict that even police forces are not allowed to gather and catalogue evidence for future reference. This paper presents an innovative solution to this problem. The authors present tools for cataloguing high- and low-level metadata of the evidence material. Furthermore, a tool for fast and accurate search for such evidence in suspects' file systems is proposed.

Keywords Live forensics · Query by example · Bees Algorithm · Child pornography

1 Introduction

Child pornography (CP) is a term that broadly describes all multimedia and writings that depict sexual activities involving a child. As at 2008, 94 of 187 Interpol member states refer to CP as a crime in their code of laws. Of these, possession of such content is punishable in 58 countries [7]. Since this crime is regarded as being extremely harmful, its prosecution is of the highest priority for police forces and law enforcement organizations around the world.

Possession of CP images is considered a crime in most countries. The law applies not only to regular citizens, but also to police units. Local regulations force police officers to destroy all evidence after the investigation is completed. This significantly complicates the gathering and presenting of evidence in police investigations. It also makes it impossible for police officers to make connections between individual cases

The presented work was supported by the European Commission, in an integrated project INDECT (Grant number 218086).

M. Grega (✉) · D. Bryk · M. Napora
AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Kraków, Poland
e-mail: grega@kt.agh.edu.pl

in order to track the distribution paths of such content. This presents a requirement for computer software to overcome this problem.

The result of the research and development is the INACT (INDECT Advanced Image Catalogue Tool) software. In the proposed solution, police officers are able to archive the metadata of the media obtained during an investigation and store it in a database using the INACT INDEXER component. When officers have access to the suspect's file system, they can use the INACT SEARCHER component to find all the media that is identical or visually similar to the samples in the database.

INACT software is classified into a set of two applications utilising the Query by Example (QbE) approach. Several such applications have already been demonstrated, including a well-known, generic MIRROR content-based image retrieval system [17], as well as more specific systems such as GAMA, designed for accessing libraries of media art [9]. The concept of using hash sets to identify suspicious files in digital forensics has been in use for a number of years, and is built into numerous forensic tools such as EnCase [6] and FTK (Forensic Tool Kit) [1]. While the above mentioned tools only allow the retrieval of images that are identical (they utilise MD5 sums), INACT also allows the retrieval of similar images, since it utilises MPEG-7 [8] descriptor values. Although both the forensic hash tools and the MPEG-7 standard are well-known techniques, their combination is novel.

The rest of the paper is structured as follows. Section 2 presents the concept of the application. It is followed by the definition of the application requirements in Section 3. Sections 4 and 5 present the INDEXER and SEARCHER components of the application respectively. The paper is concluded in Section 6.

2 Concept of the application

In order to solve the problem outlined in the introduction, a set of two complimentary applications was developed. The first application, the INACT INDEXER, is designed to be used at police stations. The police have at their disposal sets of images containing child pornography from various sources, including ongoing investigations and international channels of cooperation. Such sets are input into the INACT INDEXER. The INACT INDEXER processes the images and creates a catalogue containing information about the images (such as a description of the case in which the images were acquired) and a set of descriptors for each of the catalogued images. The descriptor set consists of MD5 hashes and MPEG-7 descriptors. The process of calculating hashes and descriptors is a one-way process, which means that the images cannot be recreated either from the hashes or from the descriptors. This allows the police to dispose of the images (as required by law) whilst retaining the information about the images themselves. The result of the INACT INDEXER analysis is a database which can be utilised by the INACT INSEARCHER application.

The database is centralised, which allows multiple instances of the INACT INDEXER to be run at the same time. This allows for country-wide deployment of the application (e.g. to all regional police forces) while retaining a coherent database of the hashes.

The INACT SEARCHER is an application designed to be used during the search of a suspect's file system. It utilises the database of hashes and descriptors created by the INACT INDEXER. The INACT SEARCHER is very straightforward to use. It

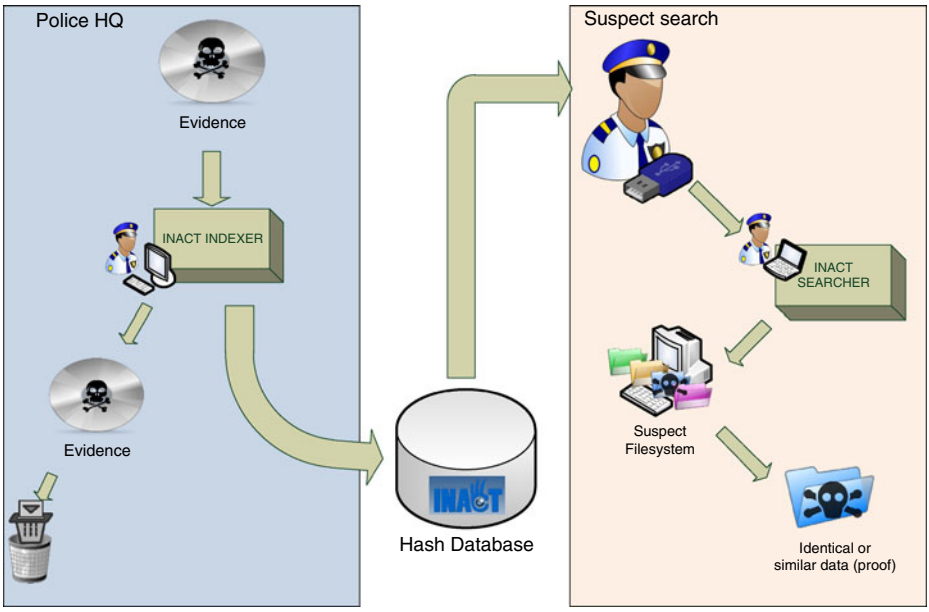

Fig. 1 INACT concept

is sufficient to select the root of a directory tree to be searched. All images that are identical (utilising MD5 sums) or similar (utilising MPEG-7descriptor values) will be retrieved and presented alongside the database information to the officer performing the search. This allows the police to draw conclusions regarding the possible sources of the images and their distribution paths. The INACT SEARCHER can be used both in a live forensic scenario and in an offline mode. The first case is addressed when the application is run on the suspect's hardware during a police operation. The offline mode is utilized when a suspect's data storage device is removed from the rest of the hardware and connected to a forensic computer via a blocker in a read-only mode. The concept of the two INACT applications is presented in Fig. 1.

3 The INACT System requirements

The INACT System intends to aid the police in combating child pornography, especially in cases where the triage model of investigation is utilised [3, 15]. In this model, a computer is initially investigated at the suspect's place of work or dwelling. It is necessary to examine a computer according to this paradigm when access to data may be irreversibly lost or significantly delayed (e.g. by powering off the suspect's computer with enabled access to an encrypted volume of a hard disk) or when time is a crucial factor to the case. This approach is also referred to as live forensics. The integrity of the original data should be assured during analysis in order to fulfil the requirements of forensic soundness, although in certain types of cases, addressed specifically by the INACT System, the need for an immediate response outweighs the benefits of legal comfort in situations in which the data is initially accessed through

a hard drive write blocker and examined thoroughly in the laboratory. However, the INACT system can be used in the latter environment as well.

The specificity of the operation explained above, and the availability of PCs to police units, impose the following general requirements on the INACT System:

1. The indexing process is performed on the PCs available to the police.
2. The search process may be performed on a computer belonging to the suspect. Their hardware and operating system may be utilised during this procedure.
3. Both the INDEXER and the INSEARCHER are intended to be operated by police officers who have only received basic training in their use. Only a basic understanding of computer usage is required from the operator.

These requirements result in restrictions to the selection of the hardware platform. In order to meet the first and second requirement, the application must be compatible with PCs based on the i486/i686 microprocessor core architecture. This architecture is widespread among users of PCs equipped with Intel or AMD microchips [16], thus it is very likely that the suspect will also be using it.

In order to meet the software configuration aspect of the second requirement, the INSEARCHER must run on different operating systems. The application must be compatible with the most popular operating systems, which are currently Windows (86.49 % of the market), Mac OS (6.54 %) and Linux (1.12 %) [11]. To do so, the Qt library is used for drawing the GUI, and the OpenCV library is used for manipulating images alongside the MPEG-7 library. The MPEG-7 library is an implementation of the tools specified in [8], written in the ANSI-C language. An SQL database is utilised for storing data locally, whilst a MySQL database provides a safe, central, remote repository. All the software solutions have been selected because they have versions compatible with the abovementioned operating systems, required to deploy a stand-alone version of the INSEARCHER. These stand-alone versions can be taken by police officers to the search location on a USB flash drive or any other mobile data storage device with a USB interface for the triage investigation. The INDEXER uses the same libraries in order to be compatible with the INSEARCHER.

The third requirement means that the GUI of both applications needs to be as simple as possible. To start the search by the INSEARCHER or the indexing by the INDEXER, police officers simply need to choose the root directory and press the start button. The third requirement also dictates that the indexing and search processes must be automated. Minimum interaction between the user and the application is required throughout, and instead files are recognised by their true format rather than by their extension. This allows officers to find images hidden in an ordinary browsing process.

4 INACT INDEXER

According to the current legal regulations regarding police investigations in numerous Interpol countries, police units cannot store evidence containing prohibited content. Any data carrier with illegal images or videos must be destroyed after it is analysed. Even if the police were permitted to gather this content, it is physically difficult to store large amounts of multimedia content.

One of the INACT project goals is to create a tool which can automatically analyse illegal files and store metadata extracted during this procedure in a database. The database should consist of information describing the content of the images (evidence). Information about the content is acquired by utilising the MPEG-7 descriptors and MD5 hashes. Calculating the descriptors is a time-consuming process. On the other hand, software used in investigations should work automatically. It should also store descriptors and hashes in a local database. As depicted in Fig. 1, in many instances police forces are able to index images simultaneously. Each of the application's instances needs to communicate with a global database and commit new records. These considerations were the main arguments for the creation of the INACT INDEXER.

4.1 Indexing mechanisms

The previewing of gigabytes of images on investigated hard disks is a tedious and time-consuming process. It is possible to miss some illegal content through this analysis. The main purpose of the INACT INDEXER is to automate police officers' work by extracting descriptor sets consisting of MD5 MPEG-7 hashes. These descriptor sets are stored in a database.

The INACT database system consist of two separate parts. The first is a local database which holds descriptors and metadata in police file systems for each instance of the INDEXER. This database can be previewed offline by police officers. The second part is a single global database used to store records from all investigations. All INDEXER instances can connect to this global database and upload collected descriptors and metadata.

The database system used in INACT is the popular SQL. There are many arguments in favour of using it:

- simple administration (also for the police units)
- SQL drivers supported in Qt4
- supported encrypted SSL connections
- supported transactions.

4.2 Indexing of movies

Illegal content can be stored not only as images, but also as video files. A video analysis system was developed for INACT the in order to index and search this type of media. It is clear that not every frame from the video file can be indexed. A movie lasting 10 min at 25 FPS (frames per second) and with resolution 640 × 320 px contains 15,000 frames. The expected time for indexing a single frame at this resolution is 10 s. This means that such a video file will be indexed in about 42 h (250 times longer than the movie itself). Another reason for rejecting this solution is that it generates a large number of non-informative frames, i.e. totally black or white (or very dark and light frames). Such frames could trash the INACT database and subsequently the search results.

The solution to this problems is to index only the informative frames of the video file. According to our definition, an informative frame is one which carries a significant amount of information; it is different from other frames, sharp, and neither too bright nor too dark. Informative frames are known as "key frames"(KF).

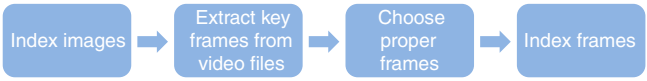

Fig. 2 INDEXER worflow

There are certain tools dedicated to the generation of KF. A good example is the *ffmpegthumbnailer*.It generates a thumbnail from a video file, which can then be used to preview the video using a service such as *YouTube* etc. Unfortunately the *ffmpegthumbnailer* and similar tools occasionaly generate non-informative frames which do not create good thumbnails and result in totally black frames. Additionally, they are not sufficiently robust to change the quality or resolution of a video file. The API for this tool could also be unavailable. These reasons have resulted in creating an internal KF extractor system.

The developed module grabs every 10th frame and calculates histogram differences [12] in order to define the initial set of potentially informative frames. The results are then filtered using an edge detector [4] in order to remove over-, under-exposed and blurred frames. The identified frames need to be checked by police officers manually in order to remove those which are in a wrong context, such as movie titles.

The INDEXER workflow is presented in Fig. 2. First, all still images in the provided evidence set are automatically indexed. Next, the KF extractor automatically generates sets of KF for all video files. Then police officers need to manually select video frames for indexing from the pre-generated KF set. Those frames are indexed in the final step. As such, police officers need to carry out just two actions during the indexing process: providing the dataset to be indexed, and selecting the frames for indexing from the KF sets extracted from the videos.

For the depicted KF, an extraction algorithm speed test was conducted. Eighteen video files of different duration and quality constituted the test set. For each of those videos, KF were generated and the time of this process was checked. The duration time to KF extraction time ratio was then calculated.

The results are presented in Fig. 3. For most videos, the calculated ratio is less than 0.1, which means that it will take approx. 10 minutes to extract KF from an average feature film.

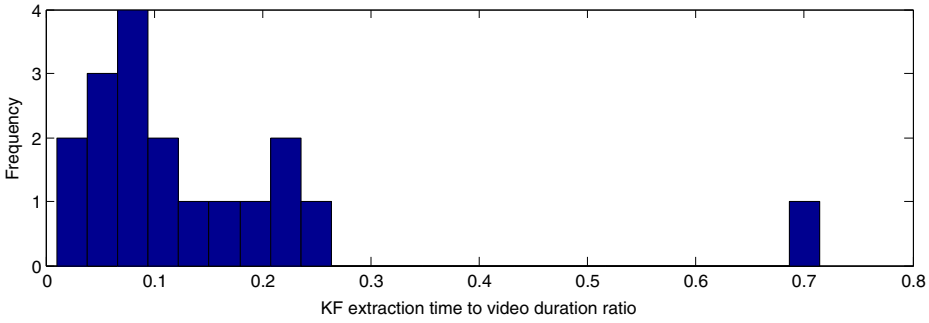

Fig. 3 KF extraction time to video duration ratio histogram

5 INACT SEARCHER

The purpose of the INACT research system is to make police work more effective. The INACT INSEARCHER (interface presented in Fig. 4) is designed and implemented to improve quality of police force operations. Special optimisation approaches were taken in order to achieve this goal.

5.1 Workflow

Let's consider a situation in which the police have a strong suspicion that a serious crime has been committed; in this instance, child abuse in the form of the collection or distribution of photos depicting juveniles in a sexual context. Today, in an era of fast Internet and cheap and widely available digital cameras, such content is stored and shared in digitalised form. This suggests that the only possible way to gather evidence is to browse through all the files on a suspect's computer and data storage devices. To do so, the police officers arrive at a site where they can access the suspect's computer, or if they previously seized the suspicious file system, they can examine it in the laboratory.

These two approaches are called online and offline forensics respectively. In case of offline forensics, the data storage device is connected to forensic equipment (a computer with special software installed) through a hardware blocker, which ensures that the contents of the data storage device remains unchanged. In online forensics, when the police officers have access to a running operating system, it is impossible to use a blocker as the device should not be powered off. Powering off a seized running device may result in encryption of previously unencrypted files. Therefore in case of online forensics all the required actions must be carried out on a "living" file system.

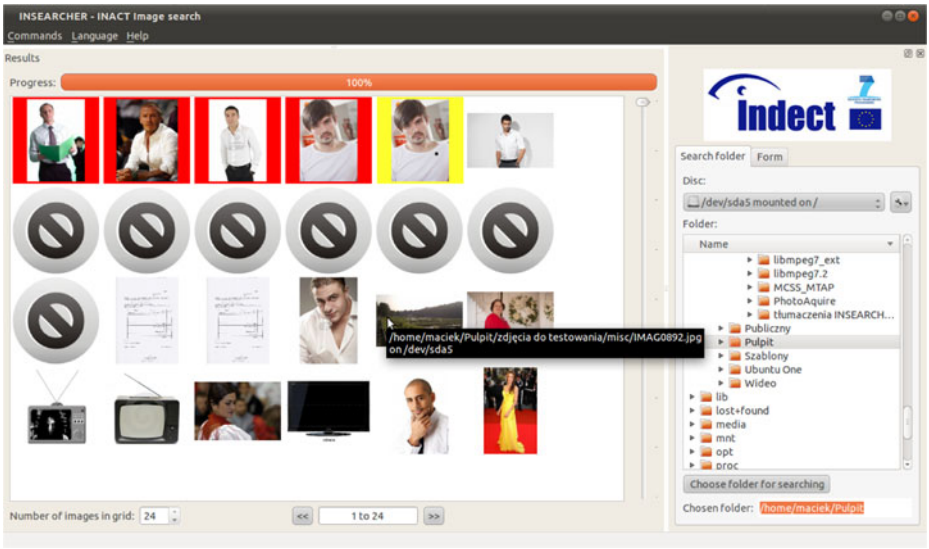

Fig. 4 The interface of the INSEARCHER application (presented with mock data)

The concept of utilising hash values, in particular those produced by cryptographic hash functions for recording and searching for illegal content, is not new. Several commercial and widely-acknowledged products for digital forensics have this functionality, although they address this issue more generally, applying this mechanism to all types of data. Only a few solutions specifically addressing the problem of searching for pornography exist. They can be divided into two groups. The first includes software such as the Forensic Toolkit with the Explicit Image Detection (EID) add-on [1], and NuDetective [5], which detects nudity on the basis of evaluated image parameters. Solutions included in the second group employ the CBIR (Content Based Image Retrieval) engine to examine images for similarities with the database. The group includes just two solutions: the Visual Search technology by ADF Solutions, and the INACT System. This class of applications has the advantage of cross-referencing ongoing investigations against previous (recorded) cases. For example, when comparing two images, the MPEG-7 visual descriptors (applied in INACT) "sense" image details such as matching backgrounds, similar camera settings, lighting, framing, and fixing on certain colourful objects (clothes, objects, environment, etc.). As a result it can produce associations between the given image and others originating from the same or similar photographic session. The Visual Search technology by ADF Solutions [2] performs a search process which allows for a similarity search based on the wavelet transform. In contrast to the INACT System, the approximate image matching engine is based on MPEG-7 standard descriptors. Additionally, the latter implements special algorithms to accelerate the search process. The INACT INDEXER analyses movies, and the INSEARCHER will gain video support in the near future. The proposed application is available to and developed in collaboration with Polish police officers. Until recently, searches were performed by police officers who manually browsed through file systems looking for images with prohibited content. The success of this process is highly dependent on the police officers' experience, accuracy, knowledge, usual location of such files, and even luck. The INSEARCHER is proposed as a solution changing the entire approach to the search process.

The exact reasons and formats of search procedures are regulated by each country's legal system; however, when police officers are equipped with the INACT System, the process is conducted differently while still in accordance with the required regulations. Firstly, the database (created with the INDEXER) on the mobile data storage device is updated with the INSEARCHER instances for online forensics, or the local data repository on a forensic station is updated for offline forensics. Secondly, police offices arrive at the site where the suspect's computer is located. They plug in a mobile storage device containing the software (in case of online forensics) or connect the disk with the suspect's file system through a hard drive write blocker to a forensic station (when the triage investigation model with offline forensics is applied). The operator runs the INSEARCHER, selects a folder, and initiates the search process. After a while, the search results are displayed in an easily understandable form. They are presented as a list, where exact matches with the database in terms of MD5 values are listed at the top and highlighted in red. Images with a 0 distance value of the descriptors are listed below and highlighted in yellow. Similar images are listed next, ordered by similarity to the images in the database. The operator can access database information on the images, preview any of the listed images in a separate window, and see the connections between them.

These functions allow the INSEARCHER to meet the second and third general requirements for the INACT system.

The above description provides additional specific requirements for the INSEARCHER application. Firstly, the operator needs to see the search results in real-time. Just a single positive search result is required for the search to be successful. Once a match is found, the search can be aborted; it is possible to press charges based on a single match. Secondly, according to the current legal regulations regarding investigations, interventions must be as quick as possible. The search must be as non-disturbing to the suspect as possible.

5.2 Search mechanism

Search duration is crucial, since conclusive results must be obtained before police intervention ends. The search process must not significantly affect the entire intervention time in order to meet the second specific requirement for the INSEARCHER. Any heavy computation could affect GUI responsiveness, which would contravene the first specific requirement for the INSEARCHER. The Colour Structure descriptor is the selected visual descriptor. It is an extended colour histogram in the HMMR colour space with human perception adjustments [10].

5.2.1 Quality of retrieval of downscaled image

The search process described above can be improved further, in particular in order to reduce intervention time. This can be achieved by applying two optimisation approaches. The first is the research on the effect of providing downscaled images to the Content-Based Image Retrieval engine on retrieval quality. The second is the application of a similarity-based search algorithm in tree structures (folder structure).

The main reason for downscaling images before submitting them as a query to the Content-Based Image Retrieval mechanism is very straightforward. If a picture of a scene is taken at 10 megapixel and 2 megapixel resolutions, in most cases its overall perception will be the same. Most people will perceive both pictures as identical or very similar when zoomed (with more details in bigger images). Images with erotic content have certain specificities, such as the frame capturing most of a silhouette, the same background if the photos come from the same session, etc. On the other hand, since the Colour Structure's properties are similar to a colour histogram, cropping or compressing (losing) information does not result in a dramatic change in the distance value between the transformed and original image. In this case the question is how much an image can be reduced to still be perceived by the descriptor values comparer as retaining similarity (measured features). The advantage of providing smaller images to the checking mechanism is a lighter computational load of the descriptor evaluation.

For the purpose of measuring how much the downscaling operation affects the evaluation of images with sexual content, the ANMRR measure (Average Normalised Modified Retrieval Rate) [10] has been adopted with a slight modification. In order to perform the experiment, a sample test set was chosen, including a total of 767 images with a physical size of 1,312 MB and an average size of a single image of 4.8 megapixels. The test set included a few image subsets of approx. 80 files each, depicting adults in a sexual context. Images within each subset

were regarded as originating from a single session, meaning that the same person was present in almost all of the pictures, details of clothing were similar, lighting was at an almost constant intensity, and the same background elements were present. In addition to the images in the series described, miscellaneous natural images with various content were present.

In order to calculate the NMRR (Normalised Modified Retrieval Rate), the following scheme has been utilised. An image from one of the series was chosen and added to the database as its only element. Other images were ordered by their distance from the single-element database. For the purpose of determining a ground truth set, several top images were chosen. They numbered between 20 and 25, depending on how many of the lowest-rated images in this range were similar to each other. Human evaluation of the content was brought in at this stage, but unlike in the original ANMRR, it had an auxiliary role. Next, images from the test set were downscaled without any filtering to several fixed sizes in megapixels, and the NMRR using the determined ground truth set was evaluated. The process was repeated for several other images, and the results for the same size in megapixels were averaged to produce the ANMRR curve depicted in the Fig. 5.

The ANMRR curve is drawn as a dashed line. The ANMRR equal to 0 means the best retrieval quality, whilst 1 means the poorest. For variously downscaled images, the total time to check the entire test set was recorded. The time consumption curve is drawn as a solid line. The graph shows that approx. 75 % of the entire image evaluation time is spent on descriptor value calculations. Compared with the overall time, as evaluated experimentally, the time spent on the MD5 value evaluation is negligible. The region of interest (ROI) is marked as a dashed rectangle. The sections of the curves within the rectangle are relatively flat. The ANMRR curve is below

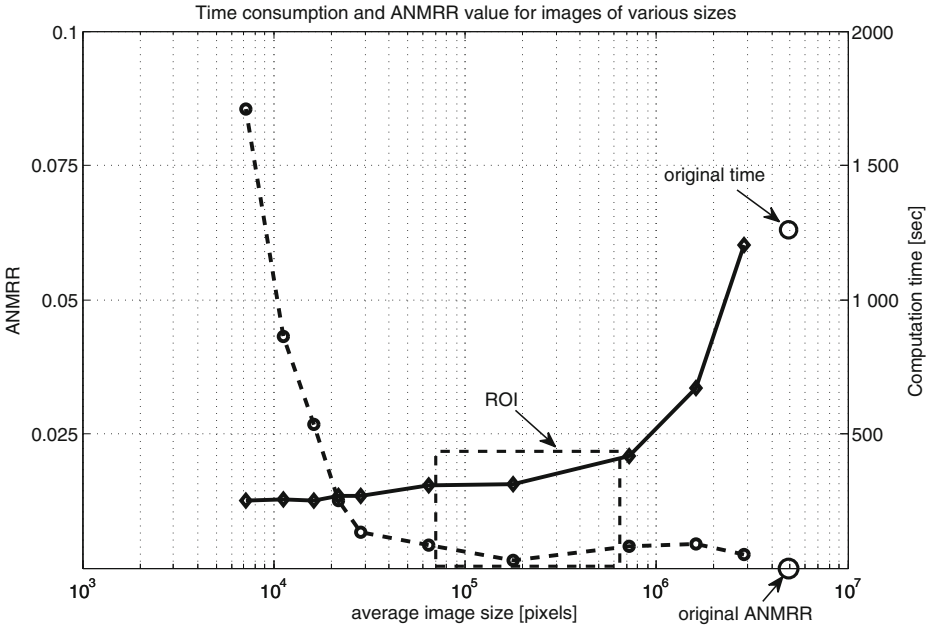

Fig. 5 Time consumption and ANMRR value for images of various sizes

0.01, and the value starts to increase rapidly for smaller images. If the obtained value below 0.01 is compared with the ANMRR value for the Colour Structure descriptor, equal to 0.14 for the 32-bit colour space quantisation (32 histogram bins)[10], the conclusion is that moderate downscaling does not introduce meaningful error to the retrieval quality for the type of images discussed. Images with smaller distances from the database can be checked further by the CBIR engine without performing the downscaling operation in order to obtain the exact distance value. Downscaling images with an approximate size of 4.8 megapixels to approximately 0.065 megapixels results in a four-fold reduction of the time spent on evaluating descriptor values for the entire test set. In [5] , the authors come to a similar conclusion that the performance of a nudity search engine based on visual parameters can be improved significantly without losing detection quality by providing a downscaled image to the classifying mechanism.

5.2.2 The Bees Algorithm—search algorithm

Considering the nature of the problem of searching for content of interest in file systems, and most possible assumptions regarding how files can be arranged by an individual, an intelligent search mechanism was proposed. It is a probabilistic algorithm inspired by an evolutionarily-elaborated food foraging behaviour of swarms of honey bees. The Bees Algorithm was previously proposed for combinatorial and functional optimisation by D.T. Pham in [13, 14].

An analogy between the problem being considered and honey bees can be drawn. Bees look for sites with the most valuable flowers. Plants that flower at the same site are of similar value for bees. They have similar genomes and grow under similar conditions. Images in the same folder are also likely to be of similar value in terms of content of interest. This is because they are likely to originate from the same photo session or to capture the suspect's preferences. Similarly to flowers, they are alike in terms of descriptor values, and they can have mutations. For example, two photos can depict the same person, firstly from the front and the secondly from the back, both against the same background. This is considered as a mutation. Descriptors are able to find similarities between the pictures despite the mutations.

Considering human habits of storing things (files), it can be assumed further that folders located nearby in the file system tree represent similar values. For bees, we can assume that sites located close to each other have similar numbers of flowers and conditions. It is possible that distant folders, similarly to flowers sites, contain different genomes (the further apart, the more different). However, the search algorithm must not get stuck in local attractors, since—as is the case in nature—valuable items can be found at distant sites (folders). In the proposed search algorithm, a superficial evaluation of the search space is performed at the beginning of the season through the random wandering of scouts with the intention to diversify search directions.

For the purpose of the algorithm, the following objects and procedures are defined:

- *bee*—pointer to a folder;
- *bee wandering*—moving a pointer randomly to a predecessor or a consequent (folder up or down in the file system structure) with equal probability for each item;

- *harvest*—random acquisition, with equal probability, of an image from a folder pointed to by a bee;
- *visited sites list*—list of folders from which an image has already been acquired (since the start of the algorithm);
- *number of scouts* (s)—number of bees reserved for the random wandering phase;
- *number of harvesters* (h)—number of bees reserved for the harvesting phase;
- *memory* (m)—a positive scalar associated with a folder, representing interest levels for the folder at some stage of the algorithm evaluation (the higher the value, the higher the levels of interest)
- *forgetting coefficient* (f)—coefficient by which the memory associated with a folder on the visited sites list is multiplied during the evaluation phase.

As feedback from the harvesting procedure, a value of the memory associated with the folder from which the image has been acquired is modified according to the following formula:

$$m' = m + \phi(d) \tag{1}$$

where: m—current value, m'—new value, ϕ—*memory excitation function*, d—distance to the database, that is the distance between the descriptor value of the query to the closest image in the database (lowest distance value).

The memory excitation function is defined as follows:

$$\phi(d) = sat\left(\left\lfloor \frac{1+P}{1+d} \right\rfloor^2\right) \tag{2}$$

where $\lfloor\cdot\rfloor$ stands for floor, P is a positive real parameter representing the minimal value of d to excite the memory, and *sat* stands for saturation function:

$$sat(u) = \begin{cases} u & \text{if } 0 \le u \le t \\ t & \text{if } u > t \end{cases} \tag{3}$$

The domain of the the ϕ function is interval $[0, \infty)$. The function itself is inspired by a function mapping input of a non-linear artificial neuron to its output. The function is equal zero in presence of a weak stimulus (when images are not similar), it starts to grow rapidly for certain threshold and saturates for a strong stimulus (when images are similar).

Three repeating steps of the algorithm can be described as:

Initial phase/Scouting phase

Each of the scouting bees (from the total population for example $s = 10$) is assigned to the root of the folder tree (folder under investigation) and it is wandered random number of times until a harvest command is issued to it. The folder to which it is pointing is added to the visited sites list. When the process is completed, the algorithm moves to *the evaluation phase*.

Evaluation phase

The visited sites list is sorted according to the value of the memory in descending order and the harvesting bees (performing the harvest operation) are assigned to the folder elements at the top. More bees are recruited for higher ranking folders (for example the corresponding proportions may be 8:8:4:4:4:2 if $h = 30$). Next, the value of *the memory* for each of the folders on the visited sites list is multiplied by the forgetting coefficient (f for example equal 0.7).

Harvesting phase

The harvesting operation is performed by each of the bees assigned during *the evaluation phase*.If the image list is empty for the current folder, the wandering operation is performed until a non-empty folder is encountered. When the harvesting operation has been performed by all bees, the algorithm moves to the scouting phase. *the scouting phase*.

The algorithm stops when all images in the folder structure have been checked. Usually, the number of scouts should be lower than the number of harvesters. A 1:3 ratio has been used, although for larger sets a higher ratio (e.g. 1:9) may be more adequate. The forgetting coefficient regulates the strength of the positive part of the feedback loop. It should fall between 0 and 1. A value of 1 means that old memory paths have not been erased, whilst a value of 0 means that memory is not sustained between cycles.

The behaviour of the algorithm can be compared to strategies observed during mining of resources. Random probing of an area is performed first. Digging starts at the most promising spots. Increasing numbers of workers are sent if excavations at these spots bring good results. A feedback loop, resulting in accelerated harvesting, is created. Meanwhile, random probing continues to evaluate other sites. A part of the positive feedback is continuously limited by forgetting, therefore if no new items of interest are encountered, the link to the current location is weakened and workers move to other promising locations. Interest in the locations is regulated. The positive proportional part assigns more workers if new resources are found from cycle to cycle, whilst the negative integrating part calms this enthusiasm over time.

For the purposes of algorithm evaluation, a series of experiments was conducted. The test set was composed of photographic sessions presenting adults in a sexual context, as well as some non-related natural images. The database contained two images from those sessions. The number of iterations to find the first and second item in the database was recorded. Evaluation of the algorithm revealed superior performance over the linear search algorithm, for which it required on average 50% of the iterations to find the image. The Bees Algorithm requires 2–5% of the iterations on average to find the first item, and 4–19% of the iterations to find the second item. The percentage of iterations required to find the second image shows that the algorithm does not stick in the first minimum. In addition, it reduces the spread of numbers of iterations required to find the first and the second image. That is, if the probability of finding an item falls between 0–100% of iterations for the linear or random search algorithm, the results for the Bees Algorithm are more dense around the given figures.

5.2.3 Overall performance of the INSEARCHER application

In order to learn the overall performance of the INSEARCHER application, an experiment on a sample test set including 7,151 images of a total size of 11,227 MB was conducted. The database contained two images, each from a different set. The following results were acquired utilising the previously mentioned desktop PC with an Intel Core 2 Duo T5600 1.83 GHz (two cores) chip using the Ubuntu 11.04 operating system:

Average percentage of iterations to find the first image: 3.93 %
Average time to find the first image: 1 min 47 s (for linear or random search: 13 min 40 s)

Average percentage of iterations to find the second image: 18.58 %
Average time to find the second image: 9 min 22 s

The times presented above were achieved using downscaling optimisation. The images were reduced in size to approx. 0.065 megapixels. Considering that downscaling images from 5 megapixels to approx. 0.065 megapixels accelerates the search process four-fold, and the proposed search algorithm reduces the average time to find the first item from approx. 50 % of iterations to 5 % iterations, then, as the result of the presented optimisation approaches, if the assumptions are met, the overall time to find the first item is reduced on average 40 times (or 4 times in the worst case).

Possible alterations to the proposed algorithm include presetting the memory value for certain folders (e.g. temporary Internet files folder) before the start, or providing a mechanism for increasing the memory value associated with a folder by marking an object of interest in the GUI display area during the search (interaction with the operator).

The usefulness of the algorithm itself is not limited to the CBIR mechanism based on descriptors, but it is a general mechanism for similarity-based searches in tree-like or net-like structures.

6 Summary

This paper introduces a novel, advanced image cataloguing tool. INACT allows for the creation of a database of images whose storage is prohibited, and an automated search for those images in a suspicious file system, as well as the creation of links between previous and ongoing cases. INACT should provide better protection for the victims, greater effectiveness in capturing the offenders, and, finally, less tedious work for police officers.

Implementation of the INDECT system based on the INDEXER and INSEARCHER modules makes it possible for police units to set up a database holding data on child pornography. The concept of storing information on child pornography images is not new. The basis for identifying images with child abuse material is a hash value calculated using a dedicated file. It is the simplest yet not very progressive method of investigating such images. The main advantage of the INDECT system is enabling police forces to search for files resembling each other in their content, based on MPEG-7 descriptors. This functionality should allow the police to expand their investigation and unveil more evidence.

References

1. AccessData: Forensic Toolkit (FTK) computer forensics software. http://accessdata.com/products/forensic-investigation/ftk. Accessed 30 Sept 2011
2. ADF Solutions: The Forensic Triage Company. http://www.adfsolutions.com. Accessed 30 Sept 2011

3. Casey E et al (2010) Handbook of digital forensics and investigation. Elsevier/Academic Press
4. Nadernejad E, Sharifzadeh S, Hassanpour H (2008) Edge detection techniques: evaluations and comparisons. Appl Math Sci 2(2–31):1507–1520
5. Eleuterio P, Polastro M (2010) Optimization of automatic nudity detection in high-resolution images with the use of NuDetective Forensic Tool. In: Proceeding of the fifth international conference on forensic computer science—ICoFCS 2010, Brasilia, Brazil
6. Guidance Software: Computer forensics solutions, digital investigations, e-discovery. http://www.guidancesoftware.com/. Accessed 30 Sept 2011
7. International Center for Missing and Exploited Children (2006) Child pornography: model legislation and global review, 6th edn. http://books.google.pl/books?id=8iqqOwAACAAJ. Accessed 30 Sept 2011
8. ISO/IEC (2002) Information technology—multimedia content description interface. ISO/IEC 15938
9. Ludtke A, Gottfried B, Herzog O, Ioannidis G, Leszczuk M, Simko V (2009) Accessing libraries of media art through metadata. In: International workshop on database and expert systems applications, pp 269–273
10. Manjunath BS, Salembier P, Sikora Th (2002) Introduction to MPEG-7 media content description interface. Wiley, Chichester
11. Operating System Market Share (2011) http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=8. Accessed 30 Sept 2011
12. Pardo A (2006) Probabilistic shot boundary detection using interframe histogram differences. In: CIARP'06: Proceedings of the 11th Iberoamerican conference on progress in pattern recognition, image analysis and applications. Springer-Verlag, Berlin, Heidelberg, pp 726–732. doi:10.1007/11892755_75
13. Pham DT, Ghanbarzadeh A (2007) Multi-objective optimisation using the Bees Algorithm. In: Innovative production machines and systems virtual conference
14. Pham DT, Ghanbarzadeh A, Koç E, Otri S, Rahim S, Zaidi M (2006) The Bees Algorithm—a novel tool for complex optimisation problems. In: Proceedings of the 2nd international virtual conference on intelligent production machines and systems (IPROMS 2006). Elsevier, Oxford, pp 454–459
15. Rogers MK, Goldman J, Mislan R, Wedge T, Debrota S (2006) Computer forensics field triage process model. In: Conference on digital forensics, security and law
16. Shilov A (2010) Microprocessor market grows moderately in the third quarter. http://www.xbitlabs.com/news/cpu/display/20101111195314_Microprocessor_Market_Grows_Moderately_in_the_Third_Quarter_Analysts.html. Accessed 30 Sept 2011
17. Wong KM, Cheung KW, Po LM (2005) Mirror: an interactive content based image retrieval system. In: ISCAS (2). IEEE, pp 1541–1544

Michał Grega (Ph.D. Eng., grega@kt.agh.edu.pl) started his university education at the University of Science and Technology in Cracow, Poland in 2001. In 2006 he presented the master thesis "Trust management in ad–hoc networks", which received highest–possible grades. He received a diploma with honors. In 2010 he finished the Ph.D. course at the Department of Telecommunications, University of Science and Technology. In 2011 he defended his PhD thesis titled "Performance

analysis of a query by example image search method in peer to peer overlays". In 2005 he joined a research team at the Department of Telecommunications at the University of Science and Technology, where he took part in several national and European projects. In 2008 he was working as a reviewer of the FP7 STREP projects on behalf of the European Commission. He is an author of over 40 publications including 5 journal papers and 3 book chapters. His research interests include, but are not limited to the aspects of multimedia in P2P overlays, multimedia search, 2D and 3D image recognition and processing and QoE estimation for multimedia services.

Damian Bryk (Eng., dmn.bryk@gmail.com) started his university education at the University of Science and Technology in Cracow, Poland in 2007. In 2011 he presented the bachelor thesis "Hearing diagnosis device: audiometer". In 2011 he stated master studies in electronics. In 2009 he joined to the INDECT project at the Department of Telecommunications on University of Science and Technology. His interest include programming, embedded systems development and design of electronic devices.

Maciej Napora (M.Sc. Eng., napora.maciej@gmail.com; born 1987) was awarded M.Sc. Eng. degree in automation and robotics from AGH University of Science and Technology (Poland) in 2011 for presenting master thesis "The research on the possibility of the effective usage of modern computer means of multimedia data description for creation of applications automatising work of the uniformed services". He is part of the INDECT project research team for development of the INACT System. He is involved in Department of Telecommunications of the university as a first-year Ph.D. student.