Chapter 2

State of the Art

The state of the art of transparent protection of multimedia content and statistical microdata is examined in this chapter. For the sake of concreteness, the multimedia content being considered consists of digital images. Open issues later addressed in this thesis are highlighted and dealt with in some detail.

2.1 Steganography for multimedia data copyright protection

In electronic commerce of multimedia contents, merchants sell products in electronic format. These contents can be copied very easily and without quality loss. When multimedia contents are sold to possibly dishonest buyers that may copy and redistribute them, an intellectual property rights problem arises which forces such contents to be protected.

Copy prevention solutions have proven ineffective, so other solutions must be deployed. A failure example of one of such systems can be found in [DVD].

Copy detection is the most promising solution. It is based on hiding an imperceptible mark in the product before selling it. This mark will keep embedded in all copies and future recovery from illegal copies will allow to prove ownership of the product (*watermarking*) or trace the dishonest user who has began redistribution (*fingerprinting*). To imperceptibly embed a mark in a product, copy detection techniques use *steganography* [KP00].

Steganography is the art of hiding a secret message within a larger one in such a way that third parties cannot discern the presence or contents of the hidden message.

There are two kinds of marks, depending on the information they carry: watermarks and fingerprints:

Watermark: The mark contains information about the owner of the content it is embedded in, so all copies carry the same embedded mark. Future retrieval of this mark allows ownership to be proved.

Fingerprint: The mark contains information about the buyer who has bought a certain copy of the product [Wag83]. In this way, all copies sold to different buyers carry a different embedded mark. Later recovery of this mark from illegally redistributed copies allows the dishonest buyer who permitted redistribution of her copy to be identified.

A copy detection scheme consists of two algorithms: *mark embedding* and *mark recovery.*

A general mark embedding procedure is depicted in Figure 2.1. It consists of an algorithm that takes as input the original object X, the mark M to be embedded and a secret key K only known to the merchant, and generates the marked object X' as output.

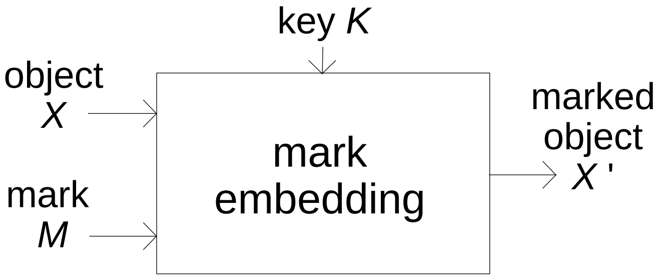

Figure 2.1: Mark embedding procedure.

A general mark recovery procedure is depicted in Figure 2.2. It takes as input a (probably) marked object $\hat{X}$, the secret key K and possibly other information depending on the specific algorithm. The procedure generates as output a Boolean value indicating whether a mark has been found or not, and depending on the scheme, the recovered mark $\hat{M}$ (when found).

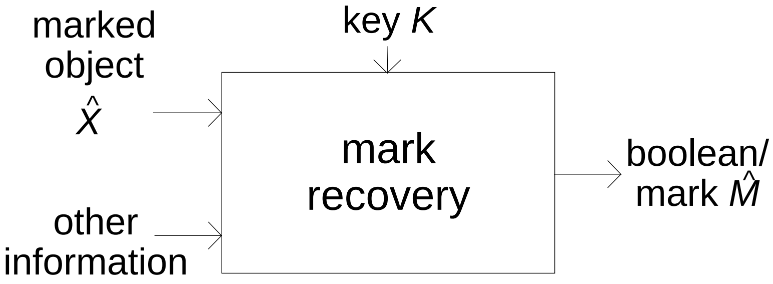

Figure 2.2: Mark recovery procedure.

2.1.1 Watermarking

In a watermarking scheme, the embedded mark contains information about ownership of the content it is embedded in. So, all watermarked copies of a product are identical.

Properties of a watermarking scheme

Next, we present a list of properties a watermarking scheme must meet:

Imperceptibility: The alterations applied to the contents for mark embedding must be imperceptible. This means the quality of content cannot be reduced after watermarking. The most appropiate metric to measure quality of watermarked multimedia contents is not yet clear. In [PA99] the Peak Signal-to-Noise Ratio, PSNR [1], is proposed as a quality metric for image marking systems. It is stated that PSNR between the original and the marked image ought to be greater than 38 dB.

Robustness: This property measures how resistant the mark is against attacks aiming at removing or making it unrecoverable. Ideal robustness is such that alterations necessary to remove the watermark without knowledge of the secret key are so large that the altered content loses its commercial value.

When dealing with multimedia contents, the amount of possible attacks is so great that it is not possible to use formal models of robustness. Instead, robustness is evaluated through benchmarks which take a marked object and apply different alterations to it. In [KP99], a benchmark is proposed to evaluate robustness of image marking systems. Its implementation can be found in [Sti]. Benchmarks like [KP99] evaluate robustness against standard signal processing attacks, but do not deal with *tamper-proofness*, *i.e.* with attacks that exploit knowledge of the algorithm internal operation. If watermarking

[1] $PSNR = 10\log_{10}\frac{255^2}{(1/L)\sum_{i=1}^{L}(X_i - X_i')^2}$, where X_i and X_i' are the original and the marked pixel values, respectively, and L is the total number of pixels of the image.

is to conform to Kerckhoff's assumption of algorithms being public-domain, tamper-proofness becomes a relevant issue.

A useful feature regarding robustness is *multiple marking* support. In schemes presenting this property, consecutive markings on the same content are possible in such a way that different marks do not interfere. In this way, individual marks can be retrieved by running the mark recovery algorithm with the corresponding key.

In a watermarking scheme, only *single-user attacks* (those performed by a single buyer) make sense. This is because all copies are identical. Later we will see that, using the fact that in fingerprinting every copy is slightly different to others, different buyers can *collude* by comparing their copies to find differences and use this information to try to compose a new copy whose mark does not identify any of them.

Information rate: This property measures how much information can be robustly embedded in a product without degrading its quality. It can be seen as the bitlength of the mark. In [PA99] it is stated that ideally one should be able to embed at least a 70-bit watermark.

Secret information: This is the minimum amount of information that must be kept secret to ensure the robustness of the scheme. Obviously, the original object must be always kept secret together with the secret key.

Imperceptibility and robustness in image watermarking

The imperceptibility property states that alterations applied to the digital object during the marking procedure must not degrade its quality.

In the case of digital images, these alterations are modifications to the color level of their pixels. A mark embedding algorithm that makes small alterations will produce a marked image with a very high degree of imperceptibility but whose mark will be easily disrupted by low intensity (and also imperceptible) attacks. We conclude that mark embedding algorithms producing small alterations to pixels usually produce imperceptible but weak marks while those producing large alterations generate robust marks but higher quality degradation. Thus, there is tradeoff between robustness and imperceptibility inherent to the marking process. To optimize this tradeoff, information is needed about the maximum increment/decrement each pixel can accomodate without generating a negative visual impact; the goal is to determine the strongest mark that can be embedded while keeping an acceptable imperceptibility.

In [Her00], the JPEG [NG96] lossy compression algorithm is used to estimate the maximum modification applicable to each pixel. The original image X is compressed using the JPEG algorithm at a given quality q. Then, the image is decompressed and gives a slightly different image X' as result. The maximum color modification a pixel i can accomodate is given by $\delta_i = |X_i - X'_i|$.

Other proposals embed information in a transformed domain. In these proposals, alteration is applied to the transformed coefficients. In this case, it is necessary to determine which coefficients can better accomodate information without degrading the image that is recovered by applying the inverse transform.

- In [HRPP$^+$98], information is embedded into the 30% middle-frequency

DCT coefficients.

- The scheme presented in [CLMT00] operates in the wavelet transfomed domain and obtains a masking function from sub-band decomposition. This masking function accounts for luminance sensitivity, spatial activity and sub-band orientation.

In Section 3.1, a new algorithm is proposed which is built over the idea that dark pixels and those in non-homogeneous regions can accomodate alterations without generating a negative visual impact.

Classification of watermarking schemes

In [Her00], watermarking schemes are classified depending on the additional information needed by the mark recovery algorithm (see Figure 2.2):

Private watermarking: The mark recovery algorithm of these schemes requires the following inputs: the probably marked object $\hat{X}$, the original object X, the secret key K and the embedded mark M. The output is a Boolean *true/false* value indicating whether the mark M has been found in $\hat{X}$ or not. The schemes presented in [CKLS97, KH97] fall in this category.

Semi-public watermarking: In these schemes, less information is required. There are two possibilities:

- The probably marked object $\hat{X}$, the secret key K and the mark M are needed. The output is Boolean and indicates whether the mark M has been found in $\hat{X}$ or not. [RA00, CLMT00, LM00, LM01] are some examples of image watermarking schemes falling in this category.

- The probably marked object $\hat{X}$, the secret key K and the original object X are needed. If a mark is found, it is given as output; otherwise, a *no mark found* message is given. The schemes proposed in [RP98, LPZ99, Her00] fall in this category.

Public watermarking: These schemes only require the probably marked object $\hat{X}$ and the secret key K. If a mark is found, it is given as output; otherwise, a *no mark found* message is given. Examples of such schemes are [HG98, AM00, Che00].

Section 3.2 presents a new semi-public image watermarking scheme offering new properties.

The concept of *oblivious watermarking* is also found in the literature. Oblivious watermarking schemes are those in which the mark recovery algorithm does not require the original unwatermarked object. Thus, from the above classification, oblivious watermarking includes both public schemes and the subset of semi-public ones that require knowledge of the embedded mark but not of the original object.

Oblivious watermarking

Oblivious watermarking offers a greater organizational flexibility and is better adapted to distributed copy detection than non-oblivious watermarking. For example, it enables the merchant to delegate copy detection to a set of agents distributed over the Internet, who can recover marks from intercepted redistributed content without having been entrusted with the original content. Such an arrangement minimizes disclosure of the original unprotected content (which stays only known to the merchant) and also minimizes storage requirements (agents do not have to store the original version of all digital content they can come across of).

Commercial oblivious watermarking schemes surviving a broad range of manipulations (*e.g.* Digimarc [Dig]) tend to be based on propietary algorithms not available in the literature.

There are two shortcomings affecting oblivious watermarking systems in the literature:

- Many published oblivious proposals require the embedded sequence to be given as an input to the mark recovery procedure (semi-public schemes).
- To our best knowledge, no oblivious proposal in the literature embeds marks so that they can survive scaling and/or geometric distortion attacks.

Next, we explain the two shortcomings above in more detail.

Many published oblivious schemes require previous knowledge of the embedded sequence for mark detection. This requirement makes mark recovery more robust but less flexible as the merchant needs to know beforehand which sequence she is looking for. Such knowledge is definitely unrealistic if watermarking is used for fingerprinting (where the merchant embeds a different serial number or buyer ID in each copy being sold).

Examples of schemes presenting this problem are [RA00, CLMT00, LM00, LM01]. In these proposals, the watermark takes the form of a Gaussian or binary pseudo random sequence s which is embedded in some transform domain. Let $C = \mathcal{T}(X)$, where $\mathcal{T}$ denotes some transform and C are the transform coefficients of the original unmodified image X. A subset c of C is modified to embed the watermark. Let $C = c \cup \bar{c}$, where $c \cap \bar{c} = \emptyset$. Denoting by $\mathcal{E}$ the watermark embedding function, the overall embedding operation can be expressed as

$$C = \mathcal{T}(X) \qquad c' = \mathcal{E}(c, s) \qquad C' = c' \cup \overline{c} \qquad X' = \mathcal{T}^{-1}(C')$$

Let $\hat{X} = X' + N$ be the image in which the presence of the watermark is tested, where N is some noise that can appear between mark embedding and mark detection. The detection operation can be expressed as

$$\hat{C} = \mathcal{T}(\hat{X}) \qquad \hat{s} = \mathcal{D}(\hat{c}) \qquad s_d = \frac{s^T \hat{s}}{\mid s \mid\mid \hat{s} \mid}$$

where $\mathcal{D}$ is a detector function and s_d is the detection statistic $(-1 \leq s_d \leq 1)$ which is a measure of the normalized correlation of the embedded and detected sequences. In these schemes, an image $\hat{X}$ is considered to contain the watermark s if s_d is greater than a fixed threshold. Of course, computing s_d requires previous knowledge of s.

To our best knowledge, robustness against scaling and geometric distortion attacks is not achieved by any published oblivious scheme. Some previous schemes assume that such attacks can be undone prior to mark recovery [RA00, LM00]. Undoing them requires knowledge of the original image which turns those schemes into non-oblivious ones.

In [AM00] an iterative search technique is used to cope with geometric attacks. The search technique seeks to emulate the inverse operation of the attack. It consists of running the mark recovery algorithm after trying various inverse operations until the bit error rate of the hidden bits drops dramatically or a high correlation with the original watermark is obtained. Thus, even if this system does not generally require knowledge of the embedded mark for recovery, it definitely requires such knowledge in order to survive geometric attacks. Furthermore, the search technique used to undo such attacks is too expensive and cannot be applied to random distortion attacks where the inverse operation is unknown.

In other oblivious schemes, geometric attacks are left for future research [CLMT00, LM01] or are not even mentioned [HG98, Che00].

Section 3.3 presents a new oblivious image watermarking scheme overcoming both mentioned drawbacks.

2.1.2 Fingerprinting

In a fingerprinting scheme, the merchant embeds a different buyer-identifying mark in each copy being sold [Wag83]. Later recovery of this mark from an illegally redistributed content allows identification of the buyer the original copy was sold to.

Properties of a fingerprinting scheme

Fingerprinting schemes are built over a robust watermarking system. This means that all properties listed in Section 2.1.1 are also necessary in fingerprinting. Some more properties must be met:

Security against collusion: The fact that every buyer receives an object with a different mark makes *collusion attacks* possible. In these attacks, a group of dishonest buyers compare their copies to find differences. This information is then used to try to compose a new copy from which none of their marks can be retrieved. Robustness against these attacks must be provided.

In [BS95], the *marking assumption* is introduced which states that, in a collusion attack, only *detectable positions* of the mark are alterable. Detectable positions are those in which the colluders find some difference when comparing their copies. The underlying watermarking scheme is assumed to be ideally robust against single-user attacks.

Security for the buyer: An honest buyer has to be sure she will not be acused falsely by a dishonest merchant. In the fingerprinting paradigm introduced in [Wag83], the mark is embedded into the content by the merchant who later sells the marked copy to the buyer. In this way, both the merchant and the buyer know the marked copy. Schemes operating in this way are classified as *symmetric schemes*. The main problem of such schemes is that a dishonest merchant can redistribute himself a copy recently sold and accuse the buyer of illegal redistribution. This argument can be used by a dishonest buyer who can claim it was the merchant who redistributed her copy.

To prevent this situation, only the buyer must know her marked copy. Schemes offering this property are called *asymmetric schemes*. In asymmetric schemes, the mark embedding procedure is replaced by a protocol in which both the merchant and the buyer play an active role. As a result of this protocol, the buyer gets a marked object to which no one else, including the merchant, has had access.

Security for the buyer also depends on security against collusion in the sense that a coalition of dishonest buyers must not be able to generate a new marked object whose mark accuses an honest buyer.

Anonymity: In schemes satisfying this property, the identity of honest buyers must be kept secret unless they act dishonestly. This means a particular buyer's anonymity will only be lost if a copy purchased by that buyer is found to have been illegally redistributed.

In [DH98], an anonymous and asymmetric fingerprinting scheme is presented.

Binary collusion-secure fingerprinting codes

In [BS95], a general construction is given for obtaining binary fingerprinting codes secure against collusions of up to c buyers (*c-secure* codes). For N possible buyers and given $\epsilon > 0$, $L = 2c\log(2N/\epsilon)$ and $d = 8c^2\log(8cL/\epsilon)$, a code with N codewords of length

$$l = 2Ldc = 32c^4\log(2N/\epsilon)\log(8cL/\epsilon)$$

is constructed which allows one of the colluders to be identified with probability $1 - \epsilon$. The authors also show that, for $c \geq 2$ and $N \geq 3$, it is not possible to obtain c-secure codes where colluders are identified with probability 1.

In [DH00b], it is shown that, for $c = 2$, collusion security can be obtained using the error-correcting capacity of dual binary Hamming codes. In this way, 2-secure binary fingerprinting codes are obtained which are much shorter than those obtained via the general construction [BS95] for $c = 2$.

In Chapter 4, a new construction to obtain 3-secure binary fingerprinting codes is presented.

2.2 Steganography for multimedia data authentication

The commonest way to authenticate digital contents is through public key cryptography. This is typically achieved by attaching a hash of the content (*i.e.* an image) encrypted under the sender/author's private key [RSA78]. This encrypted message corresponds to the digital signature of the content and can be authenticated by anyone knowing the signer's public key.

The drawback of having to deal with an attached message is obvious. This need can be avoided by using steganography. In this case, instead of appending an authentication message to the content, the sender embeds it as a watermark in such a way that any alteration will be detected by the receiver.

Authentication of digital contents based on watermarking is not intended to replace classical cryptographic authentication protocols. Watermarking cannot provide the same security properties enjoyed when exchanging data using a public key infrastructure. The advantage of watermarking authentication is that it is imperceptible (transparent).

We focus on a simpler scenario where a receiver has to receive authenticated digital content from a sender through an open channel that can be accessed by other third-party entities. We require third parties to stay unaware of protection, *i.e.* protection to be transparent. In this scenario, sender and receiver are supposed to share a secret key.

2.2.1 Fragile watermarking for image authentication

The use of secure *fragile watermarks* has been proposed as a means to verify image integrity without using cryptography. Fragile watermarks are those whose robustness is very low. Thus, the watermark can be corrupted by a very small distortion of the watermarked image.

The general paradigm assumes that the image can be divided into two disjoint sets: The set that determines the Message Authentication Code (MAC) and the set that will hold the MAC. It is important that these two disjoint sets do not interact, so that MAC embedding does not change the MAC itself.

An example of fragile watermarks can be found in [Won98]. A scheme is proposed where the LSBs (Least Significant Bits) of the original image are erased and replaced with the XOR of the hash of the 7 MSBs (Most Significant Bits) and a binary logo.

In [Fri02], a security analysis of fragile image authentication watermarks that can locate tampered-with areas is presented.

The drawback of authentication based on fragile watermarks is that the image will inevitably be distorted by the noise introduced during mark embedding. When protecting precision-critical images, such distortion may be unaffordable.

2.2.2 Invertible watermarking for image authentication

While most watermarking schemes introduce some small amount of non-invertible distortion to the image, *invertible watermarking* methods are such that, if the watermarked contents are deemed authentic, the distortion due to watermarking can be removed to obtain the original contents. Although invertible authentication allows recovery of the original undistorted image, it cannot be applied to every image, as shown in [FGD01a].

In this paradigm, the MAC is calculated from the whole image and embedded in an invertible manner in the image.

The first document on invertible watermarking is conjectured to be the patent [HJRS99]; however, this is no public-domain know-how.

In [FGD01b], two invertible watermarking methods for authentication of digital images in the JPEG format are presented. Both methods embed the MAC in the quantized DCT[2] coefficients of the image.

The first method is based on lossless compression. It computes the MAC as the hash $\mathcal{H}$ of the stream of DCT coefficients $D_k(i,j),\ k = 1, \cdots B$ of the JPEG image, where B is the total number of blocks in the image. Then, by seeding a PRNG[3] with a secret key, a random walk is followed through the set E consisting of the middle frequency coefficients of all blocks. While following the random walk, a lossless compression algorithm for the least significant bit of the coefficients is run. This random walk stops when the length difference between the compressed bit stream C and the number of processed coefficients is large enough to accomodate the hash $\mathcal{H}$ in the saved space. Denote the set of visited coefficients as $E_1,\ E_1 \subseteq E$. At this moment, C and $\mathcal{H}$ are concatenated and inserted into the least significant bits of the coefficients from E_1.

[2] DCT=Discrete Cosine Transform

[3] PRNG=Pseudo-Random Number Generator

Authentication is performed by following the same random walk while recovering $\hat{\mathcal{H}}$ and decompressing C. The bit-stream resulting from decompressing C replaces the least significant bits of E_1. Finally, we compare the recovered hash $\hat{\mathcal{H}}$ and the hash of the stream of DCT coefficients of the restored image. If they agree, the image is deemed authentic.

Note that the previous scheme requires the entropy of the least significant bits stream to be low enough to allow substantial compression. Such entropy is low because the random walk is applied over quantized coefficients (quantization can be seen as a pre-processing of the original image which introduces some non-invertible distortion).

The second method presented in [FGD01b] requires using non-standard quantization tables that must be included in the header of the authenticated image. It is based on modifying quantization tables so that all quantized coefficients are even. In this way, any alteration to the LSBs will be trivially invertible.

Additive, non-adaptative watermarking is claimed to be invertible in [FGD01a, GFD01]. In the claim, the existence of an "inverse watermarking operation" is postulated for a generic additive, non-adaptative method, but details are given only on how to derive such inverse operation for the spread-spectrum, frequency-based watermarking algorithm [HRPP+98].

A general algorithm for invertible authentication for images is presented in [FGD01a] (See Figure 2.3):

1. Let X be the original image to be authenticated. Compute its hash $\mathcal{H}(X)$.

2. Choose an additive, non-adaptive robust watermarking technique and

generate a watermark pattern W from a secret key K, so that the payload of W is $\mathcal{H}(X)$.

3. Use a special "invertible addition" to add the watermark pattern W to X to create the authenticated image X'.

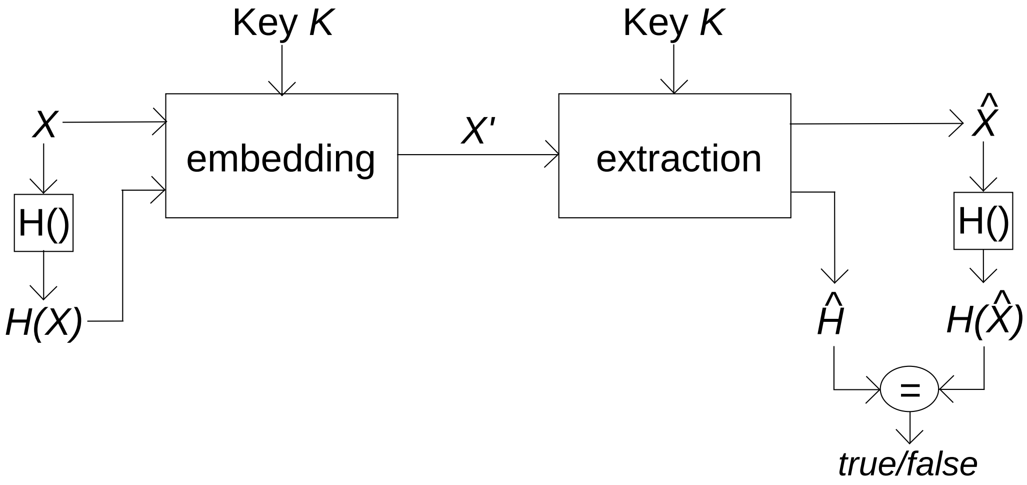

Figure 2.3: Invertible watermarking for image authentication.

The corresponding integrity verification algorithm is:

1. Extract the watermark bit-string $\hat{\mathcal{H}}$ (payload) from X'

2. Generate the watermark pattern $\hat{W}$ from the key K and the extracted bit-string $\hat{\mathcal{H}}$.

3. Using the inverse operation, substract $\hat{W}$ from X' to obtain $\hat{X}$.

4. Compare the hash of $\hat{X}$, $\mathcal{H}(\hat{X})$ with the extracted payload $\hat{\mathcal{H}}$. If they agree, the image is deemed authentic.

Section 3.5 analyzes the invertibility of the spread-spectrum watermarking scheme [HG98] and shows its applicability for image lossless authentication.

2.3 Statistical continuous microdata protection

Statistical offices must guarantee statistical confidentiality when releasing data for public use. *Statistical disclosure control* (SDC) methods are used to that end [WW01]. When data being released consist of individual respondent records, called microdata in the official statistics jargon, confidentiality means avoiding disclosure of the identity of the individual respondent associated with a published record. At the same time, SDC should preserve the informational content to the maximum extent possible. SDC methods are an intermediate option between encryption of the original data set (no disclosure risk but no informational content released) and straightforward release of the original data set (no confidentiality but maximal informational content released). SDC methods for microdata are also known as *masking* methods.

2.3.1 Current masking methods

There is a wide range of masking methods [DT01b]. From the point of view of their operational principles, current masking methods fall into the following two categories:

Perturbative: The microdata set is disturbed before publication. In this way data is altered. The perturbation method used should be such that statistics computed on the perturbed dataset do not differ significantly from the statistics that would be obtained on the original dataset.

Nonperturbative: These methods do not alter data. They produce partial suppressions or reductions of detail in the original dataset.

From the point of view of the data to which they are applied, a second classification can be established:

Continuous: These methods are suitable for masking continuous variables. A variable is continuous if it is numerical and arithmetic operations can be performed on it. Some examples are age and height.

Categorical: These methods are suitable for masking categorical variables. A variable is categorical if it takes values on a finite set and arithmetic operations on it do not make sense. Some examples are day of the week and hair color.

We do not deal here with categorical variables. So from now on, we will only refer to continuous ones.

Perturbative methods

Perturbative methods are those which mask a data set by giving perturbed values rather than exact ones.

Next we show a list and a short description of current methods:

Additive noise: This method masks data by adding noise. The simplest algorithm consists of adding white noise to the data. More sophisticated methods use more or less complex transformations of the data and more complex error-matrices to improve the results. A description of different methods can be found in [Bra02].

Data distortion by probability distribution: This method replaces values of each variable of the data set by randomly generated values following the same distribution. It consists of three steps:

1. Identification of the density function of each variable of the data set.
2. Generation of random values from each estimated density function.
3. Mapping and replacement of the random generated series in place of the confidential series.

Resampling: Let V be a variable in a data set with n records. Draw with replacement t independent samples $X_1, \ldots, X_t$ of size n from the values of V. Independently rank each sample (using the same criterion for all samples). Finally, for $j = 1$ to n, compute the j-th value v'_j of the masked variable V' as the average of the j-th ranked values in $X_1, \ldots X_t$.

Microaggregation: This method clusters records into small aggregates of size at least k. Then, each value V_i of the original data file is replaced by the average of the values V_i of the records belonging to its aggregate. Univariate methods deal with multivariate datasets by microaggregating one variable at a time. This approach is known as individual ranking. There are multivariate methods that aggregate groups of more than one variable at a time (this is in fact a parameter of the algorithm). A more detailed description of microaggregation can be found in [DM02].

Lossy compression: This method consists of regarding a numerical microdata file as an image (with records being rows, variables being columns and values being pixels). Lossy compression is then used on the image, and the decompressed image is then interpreted as a masked file. Using lossy compression algorithms with a quality parameter will allow for tunable masking intensity.

Rank swapping: First, values of each variable V_i are ranked in ascending order and information on the permutation π carried out during ranking is stored. Then each value V_i is swapped with another ranked value randomly chosen within a restricted range. Finally, the permutation π^{-1} is applied to the swapped values. This procedure is performed for all variables in the data set [Moo96].

Rounding: Original values of variables are replaced with multiples of a rounding basis b.

Nonperturbative methods

These methods rely on reductions of detail. Next we list the ones suitable for continuous microdata:

Global recoding: It consists of replacing a variable V_i by another variable V_i' which is a discretized version of V_i.

Top and bottom coding: This is a special case of global recoding. The idea is that top values (those above a certain threshold) are lumped together to form a new category. The same is done for bottom values.

2.3.2 Information loss metrics

To successfully compare different masking methods, it is necessary to have some way of measuring how much information has been lost after perturbing data.

In [DMT01], an information loss metric is proposed. Let X, X' be the original and masked data sets (both having n records and d variables). Let V and V' be the covariance matrices of X and X', respectively; similarly, let R and R' be the correlation matrices. The following information loss measures are proposed:

IL_1 describes information loss between X and X'. It computes the mean variation among both data sets:

$$IL_1 = \frac{\sum_{j=1}^{d} \sum_{i=1}^{n} \frac{|x_{ij}-x'_{ij}|}{|x_{ij}|}}{nd}$$

IL_2 is the mean variation between the averages of variables:

$$IL_2 = \frac{\sum_{j=1}^{d} \frac{|\overline{x}_j-\overline{x'}_j|}{|\overline{x}_j|}}{d}$$

IL_3 is the mean variation between covariances of variables:

$$IL_3 = \frac{\sum_{j=1}^{d} \sum_{1\leq i\leq j} \frac{|v_{ij}-v'_{ij}|}{|v_{ij}|}}{\frac{d(d+1)}{2}}$$

IL_4 is the mean variation between variances of variables:

$$IL_4 = \frac{\sum_{j=1}^{d} \frac{|v_{jj}-v'_{jj}|}{|v_{jj}|}}{d}$$

IL_5 is the mean absolute error between correlation matrices:

$$IL_5 = \frac{\sum_{j=1}^{d} \sum_{1 \le i < j} \frac{|r_{ij} - r'_{ij}|}{|r_{ij}|}}{\frac{d(d-1)}{2}}$$

All these measures are sumarized in an IL (Information Loss) measure computed as:

$$IL = 100 \cdot \frac{(IL_1 + IL_2 + IL_3 + IL_4 + IL_5)}{5}$$

The higher IL, the higher the information loss.

Computation of IL_1 implicitly assumes that there exists a one-to-one mapping between original and masked records. Corresponding records are assumed to be in the same relative position: the i-th masked record corresponds to the i-th original record.

2.3.3 Disclosure risk metrics

As pointed out in [DMT01], disclosure risk takes into account two aspects:

The first one is the record linkage disclosure risk. It is based on the assumption that an attacker has access to two datasets containing information on individual entities or people. The two datasets have some common variables that can be used to link records. A good masking record must guarantee very few records will be correctly linked.

Example: Suppose two datasets containg data about individual people. The first one contains the following variables: name, age, height, weight and other non confidential data which are publicy shown.

The second one contains medical information. The name of the respondent has been replaced by a code for anonymity's sake. Among other variables, there are height and weight.

Someone who has access to both datasets can use the common variables height and weight to link records from both sets and disclose the names of medical data set respondents (See Figure 2.4).

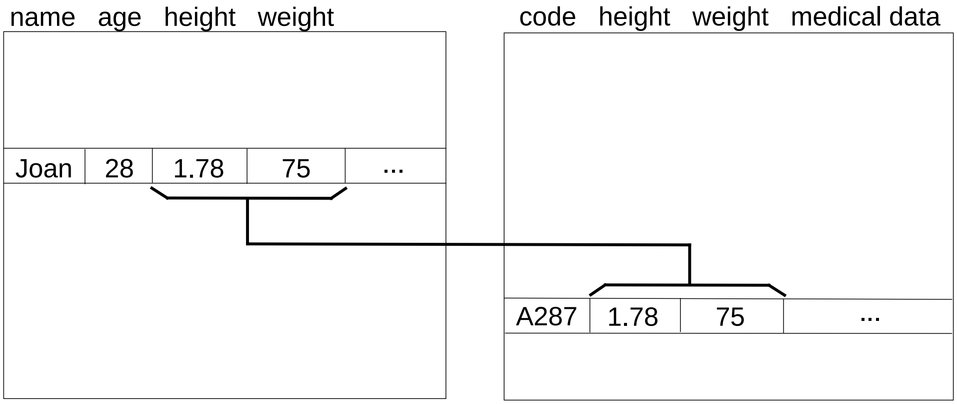

Figure 2.4: Record matching between two data sets.

The second aspect regards the information about original values an attacker can deduce once she gains access to masked values. It is necessary that, in a masked dataset, original values are not too close to original ones; otherwise they will be approximately known.

Distance-based record linkage

Let the original and masked data sets consist both of d variables (it is assumed that both data sets contain the same variables).

Assume further that the intruder can only access i key variables of the original data set and tries to link original and masked records based on these

i variables. Linkage then proceeds by computing i-dimensional Euclidean distances between records in the original and the masked data sets (using only i key variables). The variables are standardized to avoid scaling problems. A record in the masked data set is labelled as 'correctly linked' when the nearest record using i-dimensional distance is the correponding one (i-th masked record corresponds to the i-th original record).

From the record linkage method explained above, we define the DLD-i measure as the percent of records correctly linked using distance-based record linkage with Euclidean distance when the intruder knows i key variables of the original file.

The DLD (distance-based record linkage disclosure risk) measure is computed as the average of DLD-1,...,DLD-7. If data sets have a number d of variables less than 7, DLD will be computed as the average of DLD-1,...,DLD-d.

Probabilistic record linkage

This method defined in [Jar89] uses a matching algorithm to pair records in the masked and original data sets. The matching algorithm is based on the linear sum assignment model. The definition of "correctly linked" records is the same as in distance-based record linkage. This method is attractive because it only requires the user to provide two probabilities as input: one is an upper bound of the probability of a false match and the other an upper bound of the probability of a false non-match. Unlike distance-based record linkage, probabilistic record linkage does not require rescaling variables nor makes any assumption on their relative weight (by default, distance-based record linkage assumes that all variables have the same weight).

The PLD (probabilistic record linkage disclosure risk) measure is computed in the same way as DLD but using probabilistic record linkage.

Interval Disclosure

Each variable is independently ranked and a rank interval is defined around the value the variable takes on each record. This interval has size p percent of the total number of records. Then, the proportion of original values that fall into the interval centered around their corresponding masked value is a measure of disclosure risk.

ID is then computed as the average percent of values falling in the intervals around their corresponding masked values. The average is over interval widths from $p = 1\%$ to $p = 10\%$ to each side of the masked value.

2.3.4 A score for method comparison

In [DMT01], a score is proposed to measure the tradeoff between information loss and disclosure risk based on the previous measures.

The score is computed as follows:

$$Score = 0.5 \cdot IL + 0.125 \cdot DLD + 0.125 \cdot PLD + 0.25 \cdot ID$$

The lower the *Score*, the better a masking method is.

This score assumes the i-th masked record corresponds to the i-th original record and does not deal with masked sets containing a differing number of records with respect to the original one. Section 5.1 presents a modification to overcome both mentioned drawbacks.

2.3.5 Best performing masking methods

In the comparison of [DMT01, DT01a], two masking methods were singled out as particularly well-performing to protect numerical microdata:

- Rank swapping
- Multivariate microaggregation

Section 5.2 presents a post-masking procedure to enhance performance of masking methods.