Why Watermarking is Nonsense

Cormac Herley, Signal Processing, Microsoft Research

"Four thousand holes in Blackburn, Lancashire
And though the holes were rather small
They had to count them all.
Now they know how many holes it takes to fill the Albert Hall." The Beatles.

Abstract: The ease with which early watermarking algorithms were broken has given rise to a new set of schemes that are usually robust to a wide variety of attacks. We argue that this has created an illusion of progress, when in reality there is none. In common with their predecessors most published watermarking algorithms protect all objects in a neighborhood surrounding the marked object. We point out that while this is necessary it is very far from being sufficient. To withstand adversarial attack a watermarking scheme would have to protect all valuable variations of an object, not merely ones that are close to it.

Details: To begin, let us make clear that by watermarking we mean the insertion of a mark in an object **x** to form a marked object $\mathbf{x}_m$, and the use of a detector to determine whether or not a target object **y** is a valuable version of $\mathbf{x}_m$. The detector p() must return a binary answer. Thus the goal is to contrive a method of marking and a detector such that

p(**y**) = 1 if and only if **y** is a valuable version of $\mathbf{x}_m$. (*)

We assume that the system is to withstand adversarial attacks from an attacker who has access to the detector. Let's denote by **V** the set of all **y** that are valuable versions of $\mathbf{x}_m$, and by **P** the set of all **y** for which p(**y**) = 1. For an ideal watermarking scheme these two sets are very nearly coincident. They can never be precisely coincident of course since p(**x**) = 0, i.e. the original unmarked object will never be detected, a fact that will be important below.

The vast majority of the watermarking literature addresses the problem of contriving a marking method and a detector such that

p(**y**) = 1 if and only if **y** is close to $\mathbf{x}_m$. (**)

Often the detector makes its decision by thresholding some real-valued function of y d(**y**). For example, in the simplistic case where d() is related to the distance of **y** from $\mathbf{x}_m$, the set **P** will describe a neighborhood centered at $\mathbf{x}_m$. More sophisticated schemes will often make d() depend on the distance of **y** from $\mathbf{x}_m$ in some transformed space, and allow perceptually important directions to be weighted higher than unimportant ones. This is important, since the size of the neighborhood that can be protected using a Euclidean distance is limited by $|\mathbf{x}-\mathbf{x}_m|$; i.e. a sphere of protected objects centered at $\mathbf{x}_m$ will have radius limited by the strength of the original embedding. By calculating distances in a transformed space, and/or weighting directions unequally, the set of protected objects **P** can have an elongated shape and is no longer limited to the radius $|\mathbf{x}-\mathbf{x}_m|$ along any direction other than $\mathbf{x}-\mathbf{x}_m$ (the direction that points back to the original unmarked object). If the radius of **P** along all other directions is large enough that an object on the surface of **P** is not a valuable version of $\mathbf{x}_m$ then the attacker would appear to be faced with a hard problem. To reach the surface of **P** he must either (starting at $\mathbf{x}_m$) choose an arbitrary direction and advance a large distance, or determine the secret direction $\mathbf{x}-\mathbf{x}_m$ and advance a small distance. If the watermarking scheme is well designed the former will not result in a valuable version of $\mathbf{x}_m$, and there will be no computationally feasible way of doing the latter.

A flaw in this approach is that it assumes that all valuable variations of $\mathbf{x}_m$ (i.e. the set **V**) lie in this single neighborhood. We claim that it is not clear that **V** even forms a connected set. Consider, for concreteness, an example in the case of images. In many cases flipping the image left to right will have no meaningful effect on its value, and yet the flipped version will almost certainly not lie in the neighborhood **P**. For many images rotating by 90, 180 or 270 degrees makes no difference to the value (e.g. microscope images). For many images if an attacker identifies regions in the

image such as shirts etc, and changes shirts from one color to another it has no effect on the value of the image. These are offered by way of example only: the point is that by construction there exist valuable versions of $\mathbf{x}_m$ that are unlikely to be close to it in any perceptually weighted metric. Obviously neighborhoods surrounding these objects would have to be protected also. Thus it appears that **V** includes many neighborhoods. Expanding **P** until it encompasses the neighborhoods that need to be protected is not possible, since this will include many objects that are not related to $\mathbf{x}_m$. Hence it would appear we must add to **P** neighborhoods centered at each of the transformed versions of $\mathbf{x}_m$; thus **P** becomes a union of non-connected neighborhoods.

If the defender attempts protect the union of all neighborhoods two problems arise: complexity and uncertainty.

- Complexity: suppose for a protected object a suite of M transformations, each of which can take K states exists, and every one of them maps $\mathbf{x}_m$ to a valuable version. Further suppose that they can be cascaded (e.g. flipping followed by rotation). This implies a set of M^K neighborhoods the detector must protect. It is not hard to see that in the absence of an efficient way to search them this becomes unmanageable even for modest values of M and K.

- Uncertainty: the security of the system rests on listing all possible transformations that map $\mathbf{x}_m$ to a valuable version and being sure that no other such transformation will ever be found (i.e. the defender must know the structure of **V**).

The last point is the more powerful. For a given class of objects suppose we have found transformations $T_0, T_1, T_2, \ldots, T_{M-1}$ that map $\mathbf{x}_m$ to valuable versions. Even suppose that we have a detector that protects neighborhoods around all M^K variations. For no value of M can we ever be confident that this is secure. This gives the defender the unenviable task of having to plug all the holes in his armor without any information as to how many holes there are, and with the knowledge that a single omission, once found, will prove fatal.

The argument can certainly be advanced that the decision function need not be a thresholded metric. This is true, but does not help. We have argued that at least a union of neighborhoods must be protected. The defenders difficulty is that he must not merely protect all of these neighborhoods, but must in addition be confident that he has missed none. It is hard to see where such confidence would come from.

The argument can also be made that the transformations we mention (such as flipping left to right) are inadmissible and hence no protection against them is necessary. Here we get to the heart of the matter. This is an abdication of responsibility for the hard problem (*) in favor of the tractable but irrelevant one (**). We would contend that it would be challenging to find a context in which a scheme that would not withstand left to right flips or shirt color changes would be useful. If “admissible attacks” are limited to those yielding **y** that are close to $\mathbf{x}_m$ this argument is clearly circular. For most classes of objects “close to” and “valuable version of” are not synonymous, and this fact appears to compromise essentially all efforts on this subject so far.

Conclusion: We have argued that protecting a neighborhood is a necessary, but by no means sufficient condition to protect an object. The current trend of extending the list of deformations that a watermarking scheme withstands seems misguided, since, no matter how long the list, such a scheme can never with any confidence be declared secure.

Note that we have not needed to introduce confusion attacks, mosaic attacks, or systems attacks, any one of which can also fatally compromise security. Note that we make no reference to the usefulness or tractability of data-hiding, fingerprinting or marking problems without an adversary.