Adaptive Image Watermarking using Support Vector Regression

Santosh V. Chapaneri, IEEE *Student Member*

Abstract -- **This paper studies an application of support vector regression (SVR) for image watermarking. The watermark is scrambled to improve the robustness against various attacks. Feature extraction is done using a SVR neural network from the pixel locations determined randomly by a chaotic map. The watermark is adaptively embedded into the cover image leading to minimal visual quality degradation, and the scheme is resistant to image processing attacks.**

Index Terms -- **Image processing, watermarking, support vector regression, Arnold transform, chaos**.

I. Introduction

DIGITAL watermarking is one of the main research areas of information hiding [1] aimed at image authentication, copyright protection, violation detection, etc. Most watermarking techniques proposed in the literature fall into two categories: spatial-domain and frequency-domain methods. Neural networks are suggested as an alternative approach due to their high fault tolerance and potential for adaptive training [2].

There has been considerable interest to study the applications of a function approximation technique called support vector regression (SVR). Although it has many theoretical advantages, it has only recently been applied to image processing applications. Li et al. [3] suggested the use of SVR for embedding watermarks in still images. Our work extends their scheme by embedding the watermark in an adaptive manner such that the resulting image has minimal visual quality degradation. Also, the proposed scheme is resistant against various image processing attacks, and the extracted watermark is highly correlated with the original watermark.

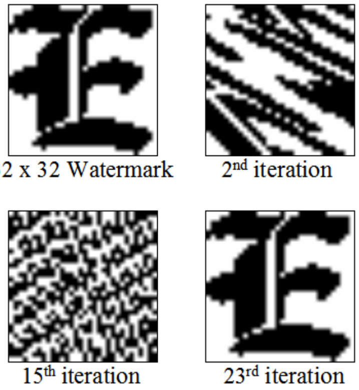
II. Preliminaries

A. Arnold Transform

A meaningful binary two-dimensional image is used as a watermark, which is scrambled using the Arnold transform [2] to improve the robustness against attacks and to enhance the secrecy of the watermark. The two-dimensional automorphism Arnold transform is given by

$$\begin{aligned} x' &= (x+y) \bmod w \\ y' &= (x+py) \bmod h \end{aligned} \tag{1}$$

where $(x',y')=$ transformed location, $(x,y)=$ original location, p = seed of permutation, w = width and h = height of image. Repeatedly transforming the image generates different results until returning to the original image due to the periodic property of transform. Fig. 1 shows the effect of the transform on a 32x32 watermark.



32 x 32 Watermark

2nd iteration

15th iteration

23rd iteration

Fig. 1 Scrambling of Watermark (p=2)

B. Chaos Map

Chaotic maps have found applications in digital watermarking to enhance security [4]. The most attractive feature of chaos in watermarking is that it is extremely sensitive to initial conditions, and two sequences generated from different initial conditions are uncorrelated statistically. Behavior in chaotic systems is aperiodic so that no variable describing the state of the system undergoes a regular repetition of values. A logistic map is the simplest chaotic map and is described as

$$X_{k+1} = \mu X_k (1 - X_k) \tag{2}$$

where $0 < \mu \le 4$ and $0 < X_k \le 1$.

When $3.569 < \mu \le 4$, the map is in chaotic state [4]. Using this mapping, pixel locations can be determined randomly from the cover image for feature learning by the neural network. The length of the sequence $X = \{X_k\}$ is set to the size of the watermark image.

C. Support Vector Regression

Support vector machines (SVM) have found application in various pattern recognition and classification problems, resulting in a maximal margin hyperplane classifier [5]. SVMs are also extended to solve regression problems using support

Santosh V. Chapaneri is a Graduate student at the Department of Electrical and Computer Engineering, University of Arizona, Tucson, Arizona, 85719, USA. Email: santoshchapaneri@gmail.com

vector regression (SVR) by the introduction of an ε-insensitive loss function. The input training data is mapped into a high-dimensional feature space nonlinearly using the kernel mapping defined as

$$Ker(x,x')=e^{-(x-x')^2/2\sigma^2} \tag{3}$$

In ε-support vector regression by Vapnik [6], the goal is to find a function $f(x)$ that has at most ε-deviation from the actually obtained targets for all the training data. The algorithm for nonlinear regression is described as

$$f(x)=\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)Ker(x_i,x)+b \tag{4}$$

where b is the bias and the Lagrangian parameters are $\alpha_i,\alpha_i^*\geq 0$. Only a few of these parameters are non-zero for which the corresponding training samples are called support vectors. The nonlinear learning problem is transformed to a linear problem in a high-dimensional space using the above mentioned kernel mapping. The regression algorithm is a result of solving the optimization problem for weights of the network:

$$\text{maximize} \tag{5}$$

$$-\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)Ker(x_i,x_j)-\varepsilon\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{l}y_i(\alpha_i-\alpha_i^*)$$

$$\text{such that}\quad \sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0,\quad 0\leq\alpha_i,\alpha_i^*\leq C \tag{6}$$

The Lagrangian parameters are obtained with quadratic programming. The constant $C>0$ measures the amount up to which deviations larger than ε can be tolerated. The bias b is found using the KKT conditions [5]:

$$\begin{aligned} &b=c_i-\varepsilon\ if\ \alpha_i\in[0,C]\\ &b=c_i+\varepsilon\ if\ \alpha_i^*\in[0,C]\\ &where\quad c_i=y_i-\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)Ker(x_i,x)\end{aligned} \tag{7}$$

The ε-insensitive loss function is attractive because, unlike other functions like quadratic and Huber cost functions where all the data points are support vectors, the support vector solution is sparse [7].

III. PROPOSED SCHEME

In our proposed algorithm, the Arnold transform is used to scramble the watermark in the preprocessing stage. A chaotic map is used to determine randomly the pixel locations where the watermark is embedded into the cover image. The SVR neural net is trained to learn the relationship between the each randomly selected pixel and its eight neighbors. Since neighboring pixels in an image are highly correlated, this correlation feature is useful to learn so that the SVR can generalize well, and so that the watermark can be recovered even in the case of various attacks on the watermarked image. The watermark is adaptively embedded into the cover image depending on the local region characteristics of the image, thus leading to minimal visual quality degradation of the watermarked image.

A. Watermark Embedding with SVR

1) Consider the cover image as I with 8 bpp grayscale of width I_w and height I_h, $0\leq I(i,j)\leq 255$, where $0\leq i<I_w$ and $0\leq j<I_h$. A binary valued watermark image is represented as W of width W_w and height W_h, $W(i,j)\in(0,1)$, where $0\leq i<W_w$ and $0\leq j<W_h$.
2) Apply the Arnold transform to W to obtain the scrambled watermark W' using (1). Convert the resulting watermark to a 1-D sequence as follows:
$$W'=(W_0^{'},W_1^{'},\ldots,W_k^{'},\ldots,W_w^{'}W_h^{'}-1) \tag{8}$$
where $W_k^{'}=W'(i,j)$, $k=iW_w+j$, $0\leq i<W_w$ and $0\leq j<W_h$.
3) Using the chaotic map (2) and a pre-specified initial value $K(=X_1)\in(0,1)$, generate a random chaotic sequence $X=\{X_k\},1\leq k\leq W_wW_h$. Divide the cover image into non-overlapping 8x8 blocks in scan-line order and label the blocks. For convenience, consider a cover image of size 256x256 having 1024 blocks. Multiply each element of X by 1024 and round towards zero to obtain a sequence of random block locations $X'=\{X_k^{'}\}$, whose range is [1, 1024] in the integer domain. Thus, $X_k^{'}$ represents the block index and we choose the second row and second column co-ordinate in each block as the selected pixel. Let S_k be the set of these randomly selected pixel co-ordinates.
4) Model the relationship between the randomly selected pixel and its eight neighbors using SVR.
The input training pattern is
$$P=(p_{x-1,y-1},p_{x-1,y},p_{x-1,y+1},p_{x,y-1},$$
$$p_{x,y+1},p_{x+1,y-1},p_{x+1,y},p_{x+1,y+1})$$
and the desired output is $d=p_{x,y}$ for all selected pixels $(x,y)\in S_k$ and $p_{x,y}$ is the intensity of the selected pixel. Thus, the training set is $\Omega=\{P_k,d_k\}$ where $k=1,\ldots,W_wW_h$. Apply Ω to train the SVR:
$$V_k=\sum_{k=1}^{W_wW_h}(\alpha_k^*-\alpha_k)Ker(P_k,x)+b \tag{9}$$
where the kernel is a radial basis function (RBF) as in (3), α_k^* and α_k are the trained Lagrange coefficients and b is the bias.
5) Embed the scrambled watermark W' into the cover image adaptively. Calculate the actual outputs V_k corresponding to P_k using the trained SVR with (9). For the selected pixel, determine the gradient

magnitude and standard deviation in its 3x3 neighborhood. Compute the embedding strength for the pixel location (x, y) as

$$e(x, y) = qI(x, y)(1+\frac{\sigma(x, y)}{A})(1+\frac{\left|\vec{\nabla} I(x, y)\right|}{B}) \quad (10)$$

where $q \in (0,1)$ is the user parameter for embedding strength, $I(x, y)$ is the pixel intensity at this location, $\sigma(x, y)$ is the standard deviation of pixel values in the local neighborhood and $\left|\vec{\nabla} I(x, y)\right|$ is the gradient magnitude computed using Sobel operator [8]. A and B are the normalization factors for the standard deviation and gradient magnitude, respectively, and can be found empirically. Embed $w_k^{'}$ by modifying the pixel values as follows:

$$I_k^{'}(x, y) = \begin{cases} V_k(x, y) + e(x, y)\ if\ \ w_k^{'} = 1 \\ V_k(x, y) - e(x, y)\ if\ \ w_k^{'} = 0 \end{cases} \quad (11)$$

where $0 \leq k < W_w W_h$. Larger q can offer better robustness but can degrade the visual quality of the watermarked image. I' is the resulting watermarked image.

The strength of the embedded watermark is adapted to the local characteristics of the image due to the influence of standard deviation (SD) and gradient magnitude (GM). As a measure of image activity, SD and GM are analogous to each other. Although experiments using only SD or GM leads to similar performance, we use both in accordance to [8]. For smooth image regions, both the SD and GM are close to zero and have an insignificant effect on the watermark embedding strength. If SD is high, indicating an image region with high variance, then the embedding strength is higher. The same case occurs with an image region with GM, indicating high gradient magnitude (candidate edge pixel). Since the human visual system is less sensitive to pixel intensity changes in busy and edge image regions, the watermarking strength can be larger in such regions without significantly affecting the watermark visibility.

B. Watermark Extraction

The watermark can be extracted from the watermarked image using the same algorithm, except that the procedure for training the SVR is not required. Using K, determine the random pixel locations as in step 3 above. Thereafter, use the trained SVR (9) to calculate the output values $V_k^{'}$ for all selected pixels. Then the watermark bits can be recovered as

$$w_k^{''} = \begin{cases} 1 \quad if\ \ I_k^{'}(x, y) > V_k^{'}(x, y) \\ 0 \quad else \end{cases} \quad (12)$$

where $0 \leq k < W_w W_h$. Apply the inverse Arnold transform, i.e. unscramble W'' to recover the extracted watermark $\tilde{W}$.

IV. EXPERIMENTAL RESULTS

Peak signal to noise ratio (PSNR) is used to assess the difference between the cover and watermarked images. To estimate the correctness of the retrieved watermark, we use the normalized correlation (NC) which is defined as follows:

$$NC = \frac{\sum_{i=1}^{W_w} \sum_{j=1}^{W_h} W(i, j) \tilde{W}(i, j)}{\sum_{i=1}^{W_w} \sum_{j=1}^{W_h} W(i, j) W(i, j)} \quad (13)$$

where $W_w W_h$ is the size of the watermark, $W(i, j)$ is the binary value of the $(i, j)th$ pixel of watermark and $\tilde{W}(i, j)$ is the binary value of the $(i, j)th$ pixel of extracted watermark, and $NC \in (0,1)$.

We tested the proposed watermarking scheme on standard grayscale cover images, such as *Lena*, *Baboon*, and *Pepper*, under various image processing attacks using *Adobe Photoshop*. The watermark used is the binary image logo made of character *E* with size 32x32. The parameter set for simulating the proposed watermarking scheme with SVR was as follows: $p = 2$ for Arnold transform, $K = 0.1564$ for chaotic sequence, $\mu = 3.7$ for mapping to chaotic state, $C = 100$, $\varepsilon = 0.008$ where C, ε are the necessary parameters for optimizing the trained coefficients and $\sigma = 10$ for width of RBF kernel for SVR. The degrading of the watermarked image depends on the watermark strength parameters q, A and B. In the experiments, we set $q = 0.1$, $A = 200$ and $B = 500$. Higher values of A and B reduce the effects of SD and GM, and higher $q(\geq 0.4)$ results in significant visual quality degradation.

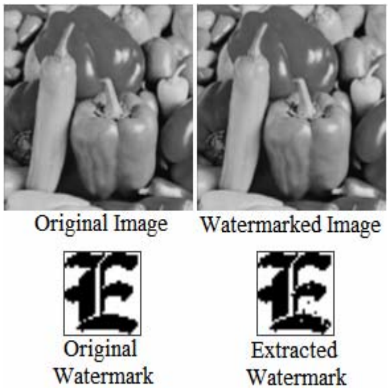


Fig. 2 Attack-free recovery for *Peppers*

Fig. 2 shows an example of a watermarked image and the extracted watermark using *Peppers*. The NC values obtained under the attack-free case were 0.9898 for *Peppers*, 0.9857 for *Lena* and 0.9795 for *Baboon* image.

TABLE I
WATERMARKS FROM LENA USING SVR

No.	Attack [9]	NC	PSNR(dB)
(a)	Attack-Free	0.9857	40.381
(b)	Sharpened	0.9262	29.855
(c)	Uniform Noise (10%)	0.9529	38.855
(d)	Liquified	0.9508	24.525
(e)	Focus Restoration	0.8873	19.792
(f)	Equalization	0.8153	19.397
(g)	Cropped (25%)	0.8043	11.504
(h)	Blurred	0.9064	35.729
(i)	Auto-Contrast	0.8627	23.053

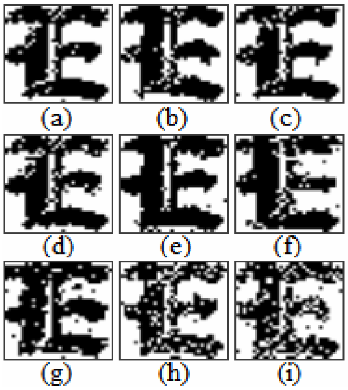


Fig. 3 Watermarks after attacks (Table I) using SVR

Table I shows a detailed analysis for the *Lena* image. The watermarks extracted from the watermarked *Lena* image are shown in Fig. 3 where the specific attacks are as listed in Table I. The extracted watermark is highly correlated to the original watermark, except in the cases of severe geometrical attack like cropping. Even with the reduction in PSNR value of the watermarked image after the attacks, the extracted watermark still remains visible and meaningful relative to the original watermark. Similar results were obtained with other standard images.

V. CONCLUSION

This work demonstrates the use of Support Vector Regression for digital image watermarking. The embedding strength of the watermark is adapted according to the local characteristics of the cover image, thus resulting in high visual quality of the watermarked image. Scrambling of the watermark with the Arnold transform improves robustness against various attacks and also enhances the secrecy of the recovered watermark. The results show that the scheme is resistant against various image processing attacks.

VI. REFERENCES

[1] J. R. H. Martin and M. Kutter, “Information Retrieval in Digital Watermarking,” *IEEE Communications Magazine*, pp.110- 116, 2001.

[2] Q. Liu and X. Jiang, “Design and Realization of a Meaningful Digital Watermarking Algorithm Based on RBF Neural Network,” *IEEE Intl. Conf. Neural Networks and Brain*, vol. 1, pp. 214-218, 2005.

[3] C. Li, Z. Lu and K. Zhou, “An Image Watermarking Technique Based on Support Vector Regression,” *IEEE Intl. Symposium on Communication and Information Technology*, vol. 1, pp. 183-186, 2005.

[4] D. Zhao, G. Chen and W. Liu, “A Chaos Based Robust Wavelet Domain Watermarking Algorithm,” *Chaos, Solitons and Fractals*, vol. 22, pp. 47-54, 2004.

[5] C. J. C. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition,” *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, 1998.

[6] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

[7] S. R. Gunn, “Support Vector Machines for Classification and Regression,” *Technical Report ISIS-1-98*, Department of Electronics and Computer Science, University of Southampton, 1998.

[8] U. Uludag, B. Gunsel and A. M. Tekalp, “Robust Watermarking of Busy Images,” *Proc. SPIE Electronic Imaging*, vol. 4314, pp. 18-25, 2001.

[9] G. D. Bouton, and B. M. Bouton, *Inside Adobe Photoshop 5*, New Riders Press, Indianapolis, 1998

VII. BIOGRAPHY

Santosh Chapaneri completed his undergraduate with distinction in Electrical Engineering from University of Mumbai and achieved first rank in the University. His employment experience includes working as DSP and Software Engineer at Patni Computers Systems Ltd, India and as a Lecturer at TCET, University of Mumbai. His special fields of interest are image analysis, image compression and video processing. He is currently pursuing MS in Electrical Engineering at the University of Arizona.