

Anti-Collusion Fingerprinting for Multimedia

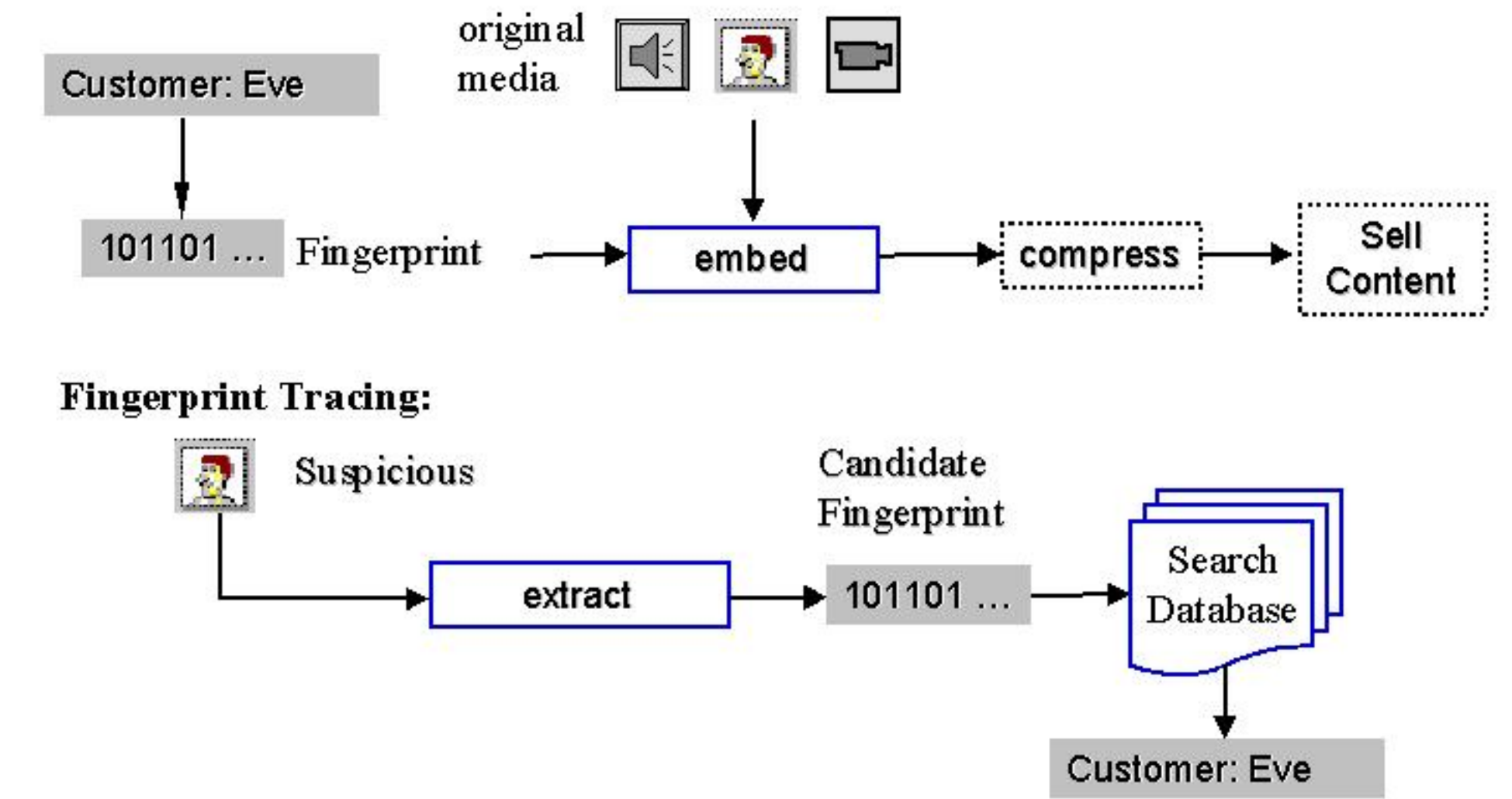
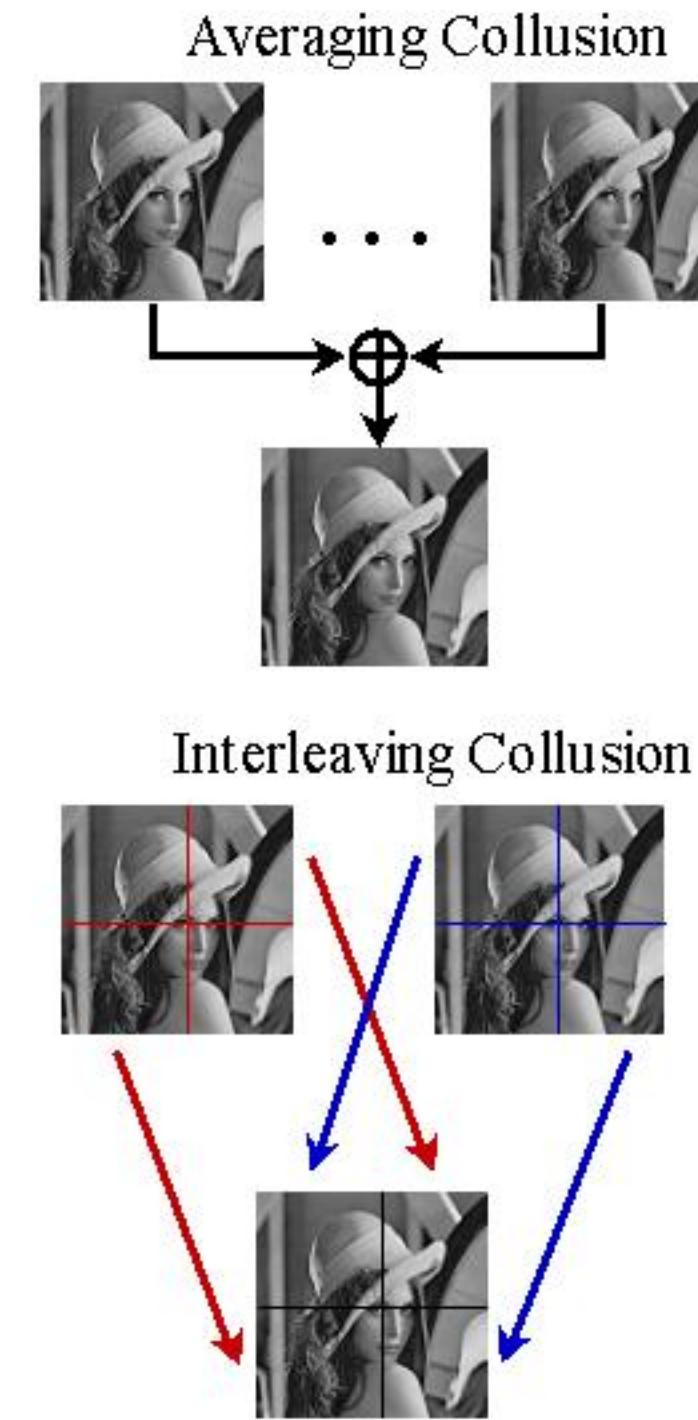
Wade Trappe

Min Wu

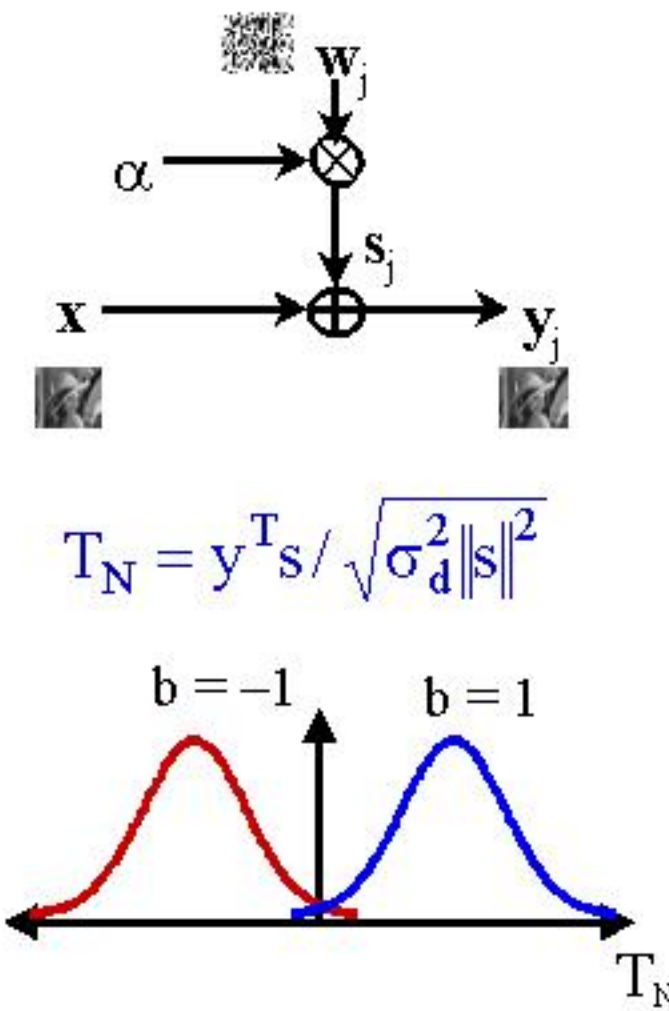
K.J.Ray Liu

Introduction

- **Encryption is useful, but not enough ...**
 - Protection by encryption vanishes after delivery and decryption
 - Customers may redistribute content in an unauthorized way
- **Robustly embed digital fingerprint**
 - Insert ID or “fingerprint” to identify each customer
 - Prevent improper redistribution of multimedia content
- **Collusion: A cost-effective attack**
 - Users with same content but different fingerprints come together to produce a new copy with diminished or attenuated fingerprints
- **Anti-collusion fingerprinting**
 - Trace traitors and colluders to actively deter collusion/redistribution
 - Rely on joint fingerprint encoding & embedding



Additive Fingerprint Embedding & Detection



- **Types of multiple-bit embedding**
 - Orthogonal modulation: $w_j = u_j$
 - Coded Modulation: $w_j = \sum_{i=1}^B b_{ij} u_i$
- **Detection via hypothesis testing**
 - Optimal detector for i.i.d. distortion $N(0, \sigma_d^2)$ is a correlator
- **Challenge**
 - Prefer short code for reliable extraction
 - Also want to trace many users/colluders

Anti-Collusion Fingerprint Codes

- **K-resilient AND ACC code**
 - A binary code $C = \{c_1, c_2, \dots, c_n\}$ such that the logical AND of any subset of K or fewer codewords is non-zero and distinct from the logical AND of any other subset of K or fewer codewords
 - **Example:** $\{(1110), (1101), (1011), (0111)\}$
- **ACC code via combinatorial design**
 - Balanced Incomplete Block Design (BIBD)

- (v, k, λ) -BIBD code is an $(k-1)$ -resilient ACC
 - Defined as a pair (X, A)
 - X is a set of v points
 - A is a collection of blocks of X , each with k points
 - Every pair of distinct points is in exactly λ blocks
 - # blocks $n = \frac{\lambda(v^2 - v)}{k^2 - k}$

Simple Example

- $(7, 3, 1)$ BIBD code
 - $X = \{1, 2, 3, 4, 5, 6, 7\}$
 - $A = \{123, 145, 167, 246, 257, 347, 356\}$

ACC code via $(7, 3, 1)$ BIBD for handling up to 2 colluders among 7 users

$$C = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

- Code length for $n=1000$ users
 - ours $O(n^{0.5}) \sim$ dozens bits
 - prior art $O(\log n)^2 \sim$ thousands

16-bit ACC Example for Detecting ≤ 3 Colluders



Detection Statistics For 3-colluder Case

