CounterFoil: Verifying Provenance of Integrated Circuits using Intrinsic Package Fingerprints and Inexpensive Cameras

Siva Nishok Dhanuskodi
University of Massachusetts, Amherst

Xiang Li
University of Massachusetts, Amherst

Daniel E. Holcomb
University of Massachusetts, Amherst

Abstract

Counterfeit integrated circuits are responsible for billions of dollars in losses to the semiconductor industry each year, and jeopardize the reliability of critical systems that unwittingly rely on them. Counterfeit parts, which are primarily recycled, test rejects, or legitimate but regraded, have to date been found in a number of systems, including critical defense systems. In this work, we present CounterFoil – an anti-counterfeiting system based on enrolling and authenticating intrinsic features of the molded packages that enclose a majority of semiconductor chips sold on the market. Our system relies on computer-readable labels, inexpensive cameras, imaging processing using OpenCV, and digital signatures, to enroll and verify chip packages. We demonstrate our approach on a dataset from over 100 chips. Our method is able to authenticate chips within 150ms, which makes it suitable for real-time use in pick-and-place machines. We show that our technique is effective and reliable for verifying provenance under a variety of settings, and evaluate the robustness of the package features by using different imaging platforms, and by wearing the chips with silicon carbide polishing grit in a rock tumbler. We show that, even if an adversary steals the exact mold used to produce an enrolled chip package, he will have limited success in being able to counterfeit the chip.

1 Introduction

Integrated Circuits (ICs) take on critical roles in today's society, but the supply and distribution channels for ICs present a large, diverse, and vulnerable attack surface. One such threat is counterfeit parts, which are a significant and increasing threat to the reliability of electronic systems. Counterfeits are defined by the US Department of Defense as "unauthorized copies and previously used parts that are made to look new, and are sold as new" [45]. Misrepresented ICs such as speed binned parts that are remarked to a higher speed grade to increase selling price [43] can also be considered counterfeits. Prior research claims that recycled and remarked chips together make up 80% of all counterfeiting incidents [20]. These types of counterfeit parts are enabled by a lack of traceability through distribution channels as parts change hands through resellers and system integrators. DARPA notes that chain-of-custody solutions are unworkable for securing distribution due to components that may change hands 15 times before final installation [30]. Our work addresses this critical security problem by giving an approach for securing parts through distribution channels without chain-of-custody.

Estimates variously place the direct losses from electronics counterfeiting at $3B-$7.5B [27], and the potential risk due to counterfeiting at $100B-$200B [41, 43]. The most commonly counterfeited electronics are said to be analog ICs, microprocessors, memories, programmable logic, and discrete transistors [20, 26]. Documented cases of counterfeit parts include purported microcontrollers that were found to be remarked voltage regulators [51], four instances of counterfeit parts in the Avionics Systems of C-27J aircraft [48], and refurbished flash memory devices in Terminal High-Altitude Area Defense (THAAD) mission computers that led to a recall of 50 systems [45].

Counterfeit parts such as these present clear security risks. However, it is important to note that these counterfeit parts are not targeted attacks against the specific systems in which they were found. Instead, the counterfeit parts are created and sold to earn profit. Their inclusion in critical systems is coincidence, due to the complicated global supply chain that allows common parts to be purchased on the market without clear and verifiable evidence about their provenance.

In this work we propose and evaluate CounterFoil, a system that uses inexpensive cameras to check intrinsic variations in semiconductor packaging as means of verifying provenance. We name our system CounterFoil both to reflect its aim of foiling counterfeits, and because the enrollment records it uses are analogs for counterfoils kept by issuers of cheques[1]. The specific contributions of this paper are as

[1] Counterfoil - "The part of a cheque, receipt, ticket, or other document that is torn off and kept as a record by the person issuing it." https://en.oxforddictionaries.com/definition/counterfoil

follows:

- We show, for the first time, that individual chip packages can be recognized and authenticated using intrinsic variations in surface features, and that even chip packages produced by the same mold can be distinguished. While there are a number of research publications that authenticate objects from unique features, ours is distinct in exploiting surface variations in molded parts.

- We present a system, based on low cost cameras, image processing, and digital signatures, that can validate provenance of chips and thereby help keep counterfeits out of systems.

- We evaluate the performance of the system with regards to authentication, runtime, tolerance to variation in lighting and magnification, and resilience against wear.

2 Background and Related Work

Strategies for preventing counterfeit parts from being used in systems can be broadly classified as either trying to detect anomalies, or else authenticating individual chip instances that are trusted.

2.1 Anomaly Detection as Counterfeit Testing

A common approach in counterfeit identification is to train a model based on a population of known good parts. When faced with a part of unknown provenance, a battery of tests is then applied and a classifier is used to evaluate its consistency with the trained model. The applied tests include physical inspection (visual [4], x-ray imaging, microblast analysis of the surface, spectroscopy, ion chromatography), electrical inspections [7,29], and checking for aging using silicon odometers [2], ring oscillators [21], dynamic current signatures in adders [57], or other circuits that change in a measurable way with use. If any tests reveal an anomaly, the part can be deemed counterfeit. Anomaly detection techniques are used as part of qualification procedures by the US Department of Defense to minimize the risk of counterfeits, but "may not definitively distinguish authentic parts from counterfeit parts" [47]. Machine learning and neural network based techniques [49] detect anomalies in microscopic features to classify genuine and counterfeit parts. Unlike these approaches our technique relies on extracting unique fingerprints from individual parts to authenticate provenance and thereby detect counterfeits.

2.2 Authenticating Trusted Parts

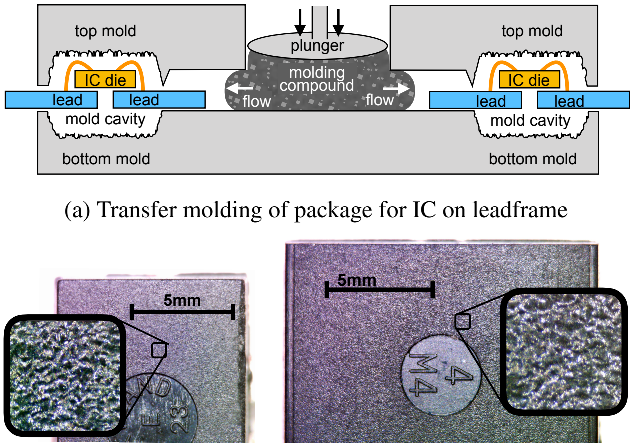
An alternative to anomaly detection is to identify and authenticate individual part instances using unique or hard to clone features. If a part is trusted at one point in time, and later a part can be validated as being the same one that was earlier trusted, then a judgment can be made that the part is still trustworthy. Non-microchip versions of this style of object authentication include human fingerprints [16], anti-counterfeiting features in currency [42], variations in surface texture of blank paper [11] and 3D printed products [32], and variations in the length of compact disc pits and lands [22]. Similarly, Physical Unclonable Functions (PUFs) are a type of physical fingerprint that can be used for authentication of parts. PUFs can be based on random delays in silicon [18], power-up fingerprints of Static Random Access Memory [19, 24, 50], randomly scattered dielectric particles in a protective coating [55], or unique Radio Frequency emissions [12,13], among many others. PUFs have also been used in conjunction with RFID-tags to detect counterfeits [54].

Several existing strategies for validating provenance of microchips are implicitly relying on the IC package as the basis for trusting the enclosed silicon die. The DARPA SHIELD project aims to embed inside IC packaging a secure dielet that can be interrogated wirelessly to validate provenance of the part [30]. A company called Applied DNA Sciences offers a botanical DNA taggant that can be applied to various goods including microchip packages [23] to support traceability through distribution. To date, working with the Defense Logistics Agency (DLA) of the US Department of Defense, the technology has marked over 700,000 microchips [38]. Both package-embedded dielets and package tagging have an underlying assumption that an adversary cannot easily swap a microchip out of its package, and therefore validating the package provenance suffices to validate the provenance of enclosed microchip. We will use this same assumption in our work which is based on packaging.

2.3 Transfer Molding for IC Packaging

Like DARPA SHIELD and DNA tags, our approach (Fig. 2) also uses the IC package as a basis for trusting parts. However, our technique exploits intrinsic features of IC packages instead of adding something to the package. We give in this subsection for reference an overview of how IC packages are created.

Transfer molding (Fig. 1) is the typical procedure used for packaging high-volume integrated circuits [6, 10]. Most DIP (dual in-line package), SMT (Surface-Mount Technology), and QFP (Quad Flat Package) packages are created this way, as well as more advanced packaging styles such as system-in-package. In the transfer molding process, each silicon die is first attached to a metal leadframe, and the pads from the die are wire-bonded to the individual leads to create electrical connections. Each leadframe-mounted die is then placed in a mold cavity, with the leads extending out the side of the cavity. A plunger liquefies pucks of epoxy molding compound using temperature and/or pressure. The liquefied compound flows through runner channels into the mold cavity to surround the



(a) Transfer molding of package for IC on leadframe

(b) Surface texture of molded packages

Figure 1: Transfer molding is the mechanism used for packaging most high-volume microchips.

Algorithm 1: ENROLLCHIP

```
Input: Image $img$ of chip surface with marker
       attached. Private key $k_{pr}$ for signing messages.
$eid \leftarrow readMarker(img)$
$f_{eid} \leftarrow extractKeypoints(img, r, \theta, w_{enroll})$
$s(f_{eid}) \leftarrow Sign(k_{pr}, f_{eid})$
$database[eid] \leftarrow f_{eid} \| s(f_{eid})$
return
```

Algorithm 2: VERIFYCHIP

```
Input: Image $img$ of chip surface with marker
       attached. Public key $k_{pub}$ to check signatures.
Output: Success or failure to verify chip as authentic
        according to the identity on its label
$id \leftarrow readMarker(img)$
$f_{eid} \| s(f_{eid}) \leftarrow database[id]$
if $VerifySignature(k_{pub}, s(f_{eid}))$ then
    $f_v \leftarrow extractKeypoints(img, r, \theta, w_{verify})$
    if $score(f_{eid}, f_v) > threshold$ then
        return $success$
return $fail$
```

die and form the shape of the package. After the compound solidifies, the molds are released, and the leads are separated from the remainder of leadframe, which is discarded. The metal leads protruding from the formed package are now the pins of the packaged chip that will connect it to a printed circuit board. Further details on the many packaging styles for integrated circuits can be found in a popular textbook on the topic [53].

Several sources of variability in transfer molding can impart unique features to a package surface. The mold has a surface roughness that gets imprinted onto the package. The surface texture of the mold changes over time as residue material accumulates on the mold, and molds require cleaning to mitigate this build up [25]. Additionally, the molding compound itself, and its curing, contribute a certain amount of unpredictability. The molding compound is an epoxy that contains a number of fillers including crushed quartz or alumina that comprise 75% or more of the compound, and provide thermal conductivity. The size of the filler particles can range from 20-100μm, and the orientation and distribution of filler particles in the package is unpredictable. The package during post-mold curing also experiences shrinkage, cracks, porosity, and voids [52]. Due to aforementioned variation sources, even chips packaged in the same mold could have differences in their package surface.

3 Description of Approach

COUNTERFOIL uses package surface features to authenticate provenance of individual chips as shown in Fig. 2. The two participants in the scheme that interact with the chip are denoted as the enroller and a verifier. The enroller acts on behalf of a chip manufacturer that wishes to sell parts with an assurance of provenance. The verifier is a customer that has purchased the chips on the market and wants to check whether they are legitimate. Both the manufacturer as enroller, and customer as verifier, have incentives for participating in the presented scheme. The chip manufacturer can make their products more attractive by offering an assurance that authentic parts bearing their branding can be verified as produced by them. Importantly, they can accomplish this without needing to trust every point in their distribution channels. The chip customer is incentivized to participate because systems that are free from counterfeit chips can avoid costly failures or recalls that are caused by counterfeits [45].

The enroller extracts fingerprints from package surface features using image processing and publishes information about enrolled chips to a public database. Integrity of database entries is assured by digital signatures. The enroller holds a private key k_{pr} for signing messages, and gives the corresponding public key k_{pub} to any parties that wish to act as verifiers. Our implementation uses the simplifying assumption of pre-existing public keys for enroller and verifier, but in practice this could, for example, rely on a trusted certificate authority. The enroller uses the private key to sign database entries when writing them, and the verifier uses the enroller's corresponding public key to check the signatures when reading from the database. More details about the enrollment (Alg. 1) and verification (Alg. 2) procedures are given below. Details of the image processing performed in enrollment and verification are deferred to Sec. 4.

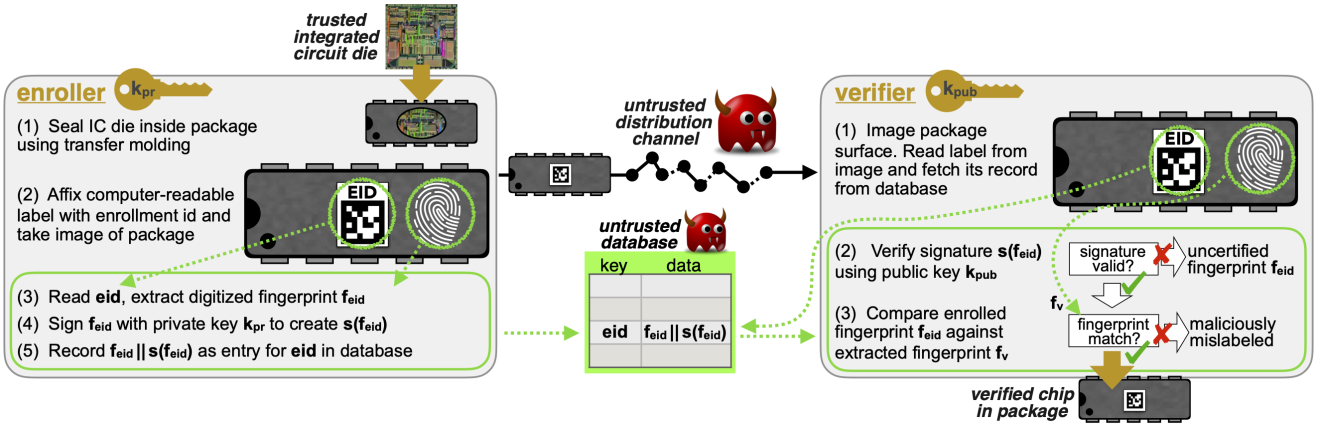

trusted integrated circuit die
enroller
k_{pr}
(1) Seal IC die inside package using transfer molding
(2) Affix computer-readable label with enrollment id and take image of package
EID
(3) Read **eid**, extract digitized fingerprint $\mathbf{f_{eid}}$
(4) Sign $\mathbf{f_{eid}}$ with private key $\mathbf{k_{pr}}$ to create $\mathbf{s(f_{eid})}$
(5) Record $\mathbf{f_{eid} || s(f_{eid})}$ as entry for **eid** in database
untrusted distribution channel
untrusted database
key
data
eid
$f_{eid} || s(f_{eid})$
verifier
k_{pub}
(1) Image package surface. Read label from image and fetch its record from database
EID
(2) Verify signature $\mathbf{s(f_{eid})}$ using public key $\mathbf{k_{pub}}$
signature valid?
uncertified fingerprint $\mathbf{f_{eid}}$
$\mathbf{f_v}$
(3) Compare enrolled fingerprint $\mathbf{f_{eid}}$ against extracted fingerprint $\mathbf{f_v}$
fingerprint match?
maliciously mislabeled
verified chip in package

Figure 2: Protocol for package fingerprinting. Trusted enroller labels each package and then enrolls it by extracting and then signing a set of keypoints associated with the package. Verifier compares the enrolled keypoints against the package to determine whether the package is consistent with its label.

3.1 Enrollment

The enrollment procedure should occur as part of the packaging of an IC. The IC should be trusted at the time of packaging, as the goal is to later tie provenance back to this point. Each die is sealed inside of a molded plastic package as usual by means of transfer molding (see Sec. 2.3 and Fig. 1). After the package hardens and cures, a label with a computer-readable identification marker is affixed to the surface of the package. The marker represents an insecure numerical identifier of the chip instance, similar to a serial number, which we denote as its *eid* (enrollment identifier). The enroller then takes an image that captures both the marker, and the package surface in the vicinity of the marker, from which the fingerprint will be extracted. A digitized enrollment fingerprint f_{eid} is extracted from the image, using a procedure that will be explained in Sec. 4.2. The date of manufacture and other metadata can be appended to the fingerprint at this point. The enroller creates signature $s(f_{eid})$ by digitally signing fingerprint f_{eid} using private key k_{pr} (Alg. 1, line 3). An entry is added to the public database to associate the identifier *eid* with $f_{eid}||s(f_{eid})$ (Alg. 1, line 4). Once the chip is enrolled to the database, it is released into distribution channels.

3.2 Verification

The verification procedure checks authenticity of chips at the end of distribution. The verifier takes an image of the chip that includes both the marker and the package surface in the vicinity of the marker. The insecure identifier (*eid*) of the marker is extracted from the image. The enrolled data $f_{eid}||s(f_{eid})$ for this identifier is accessed from the database (Alg. 2, line 2). The validity of signature $s(f_{eid})$ is checked using the public key k_{pub} of the enroller (Alg. 2, line 3). The enrolled fingerprint f_{eid} is compared against a new fingerprint f_v that is extracted from the relevant area of the chip package surface. If the similarity score exceeds a chosen threshold, then the package surface is determined to match the record (Alg. 2, line 5). The chip is verified as authentic only if the digital signature is valid, and the fingerprints match. The validity of the signature ensures that the enrolled fingerprint in the public database was created by the enroller and has not been modified. The fingerprint match ensures that the enrolled data is not being used to authenticate a chip other than the one that was enrolled, a scenario that would arise if a label was copied or transferred from one chip to another. The verification procedure is currently performed on a workbench in our lab, but could later, for example, be integrated into a pick-and-place machine at the end of distribution that picks chips from reels and places them appropriately onto printed circuit boards.

3.3 Attacker Capabilities and Security Considerations

The attacker considered in this work is a profit-motivated counterfeiter that forges chips for purpose of selling them on the market. This type of profit-seeking attacker is responsible for prior counterfeit parts found in sensitive systems, but note that it does not include nation-state attackers that may spend large amounts of money to create malicious forgeries to bring down targeted high-value systems. For a profit-seeking attacker, if the effort of forging chips exceeds the selling price of the chip on the market, there is no incentive to forge the chips. At the same time, the cost for anti-counterfeiting technology in commodity parts cannot exceed what the producer or consumer of the parts is willing to spend for the guarantee of provenance.

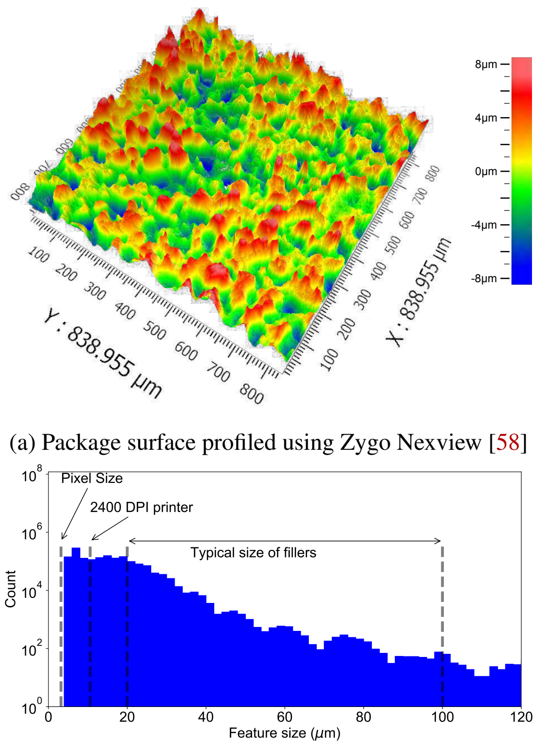
The security of our approach relies on assumptions similar to those in earlier work on certificates of authenticity [13]. Our assumptions relate to the enrollment and verification protocol, the uniqueness of package fingerprints, and the difficulty of creating forged chip packages that match legitimately enrolled fingerprints. Among these three, the first is intended to be uncontroversial, and the latter two are supported by experimental data in the paper.

1. **Protocol Integrity:** We make the standard assumption that an adversary is not able to obtain the enroller's private key or forge digital signatures without having the private key. We assume that the enroller is trusted to only package legitimate integrated circuits, and to enroll only these packages with the private key k_{pr}.

2. **Unique Fingerprints:** We rely on the fact that package fingerprints created under ordinary conditions are unique and are identifiable via image processing. Specifically, an enrolled fingerprint from one package will not be deemed a match for any package other than the enrolled one. Fingerprint uniqueness binds the enrolled data to a specific chip instance. If labels are later affixed to chips other than the enrolled, the enrollment data associated to the label will not match the chip characteristic. This prevents an adversary from successfully copying or transferring labels across chips.

3. **Difficulty of Package Forgery:** We assume, and then support experimentally, that package fingerprints are random and difficult to control. This prevents an adversary from creating a new package surface that matches a legitimate enrolled fingerprint. We support this assumption by showing that even chips from the same mold have different fingerprints. This implies that even possession of an identical mold will not enable an adversary to successfully forge packages and therefore forgery requires a more advanced manufacturing process than what industry uses for packaging chips. Regardless of the process used to create forgeries, an adversary will have to create recognizable features with sizes on the order of $10\mu m$ (see Fig. 3). Besides attempting to clone the package surface an attacker could print a label with features from a legitimate chip. However, the printing task is seemingly out of reach of many technologies such as high-end 2400 DPI printers, which have a dot size of $10.6\mu m$ and can only print reliable features at a much larger scale than its dot size. Aside from forgery, an adversary might transfer the package from a legitimate part to a counterfeit IC, but there would be no profit motive to this, as it would destroy a legitimate chip to create a single forged chip.

Practical security concerns of our prototype system warrant further discussion. One concern is that an adversary could make a chip unverifiable by removing, moving, or damaging its label. This threatens reliability more than security because it does not falsely authenticate counterfeits, and because counterfeiters would not directly profit from destroying the labels.



(a) Package surface profiled using Zygo Nexview [58]

(b) Extracted SIFT feature sizes from image processing.

Figure 3: Package surface features, and distribution of feature sizes extracted from package surface images using OpenCV.

For reliability, the paper labels used in our prototype system would likely be replaced by more robust markings when deploying COUNTERFOIL at production scale outside of the lab. A second practical concern pertains to the use of a public database for enrollment records. The records in the database reveal information about quantity and schedule of produced parts, which may be sensitive to the manufacturer. Similarly, database queries that happen in the clear could reveal business information about the consumer. Where this is a concern, the enrolled data could be made private and provided only to trusted verifiers, or cryptographic protocols for oblivious transfer [44] or anonymous credentials [9] could be used to ensure privacy.

4 Image Processing and Analysis

Our system relies on image processing as part of enrollment and verification. Enrollment generates a digitized representation of recognizable features within a selected area of the package surface. Verification later scores the record of enrolled features against a new image of the package surface. In this section we describe the computer vision algorithms used. Our algorithms are written in C++ using OpenCV [8] for the image processing.

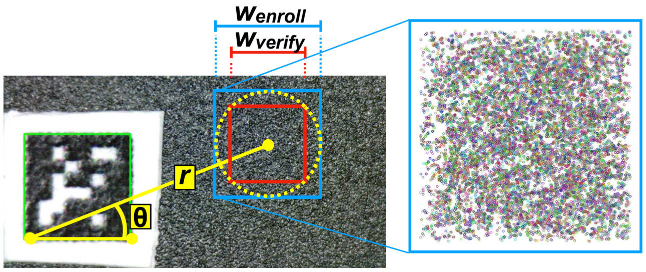


Figure 4: Image of chip with affixed marker. The position of enrollment ROI is shown by the blue box, and the callout shows the keypoints extracted from the ROI. The ROI that would be used for verification is the smaller red box. The size and position of both ROIs are defined relative to the marker, as shown by annotations in yellow.

4.1 Aruco Marker Labels and ROI Detection

Our system uses computer-readable labels (Fig. 2) to represent the purported identity of a package. The labels are placed, to the extent possible with manual placement, in the same position on each package. For convenience the labels are also used as fiducial marks to define the Region Of Interest (ROI) in the enrollment and verification images, although other easily-recognized features could be used instead of the labels for this purpose. Aruco, the specific marker system that we use, is a square-based fiducial marker system with binary codes [17]. Aruco marker dictionaries are configurable, allowing for an arbitrary marker capacity (in bits) and number of markers. We use Aruco markers to label the chips with the search tag of the public database. The four corners of the marker allow for detection of image orientation (pose estimation) which we leverage to determine the ROI for further processing. Figure 4 shows a detected marker with its top-left corner used to determine the center of ROI at a distance $< r, \theta >$ relative to the marker. Depending on whether the image is being processed for enrollment or verification, the ROI selected from the image would be either ROI_{enroll} (blue square) and ROI_{verify} (red square). Both squares are centered at the same point, and have a size that is defined relative to the marker size for magnification invariance. The width of the larger square is $w_{enroll} = 2mm$, and the width of the smaller square is $w_{verify} = w_{enroll}/\sqrt{2}$. The difference in ROI sizes ensures that the ROI from enrollment will always contain the ROI from verification regardless of rotation. Consider the yellow circle in Fig. 4 which is centered at point $< r, \theta >$. Regardless of the image orientation, the red square will always be contained within the circle, and the blue square will always contain the circle. Therefore, the blue square (ROI_{enroll}) will always contain the red square (ROI_{verify}). Further, ROI_{enroll} is chosen larger than ROI_{verify} to save runtime, as the verification involves more processing steps than enrollment. In our experiments we use $r = 5mm$ and $\theta = \pi/8$.

4.2 Feature Enrollment

The enrollment process extracts distinctive features from an image which are suitable for matching and object recognition, and stores them as compact feature descriptors. A number of well-known image processing techniques exist for feature detection and description, such as Scale Invariant Feature Transform (SIFT) [33], Oriented FAST and Rotated BRIEF (ORB) [46], Binary Robust Invariant Scalable Keypoints (BRISK) [31], and Speeded-Up Robust Features (SURF) [5]. These techniques are commonly used in applications such as image stitching, where image alignment requires finding corresponding points of objects in two different images that contain the objects. Our work is agnostic to the choice of algorithm, but based on empirical evaluation (as will be discussed in Sec. 5.2.1) we choose ORB.

We first pre-process the image (ROI) using Contrast Limited Adaptive Histogram Equalization (CLAHE) to improve the contrast and tolerance to variation in lighting intensity. We then use OpenCV's implementation of ORB to extract image features. The keypoints are detected by Oriented FAST algorithm and described by 256-dimensional rotated BRIEF descriptors [46]. Similarity between two keypoints can be evaluated using *feature distance*, which is the Euclidean distance between two keypoints in the 256-dimensional feature space. The keypoints also have associated positions within an image, and we will use *pixel distance* to denote the Euclidean distance in two dimensions between pixels in an image. For the sake of predictable runtime, we restrict the number of keypoints to $1{,}000/mm^2$ of package surface. Fig. 4 shows the keypoints extracted from the region of interest.

The enrolled features are stored in a public database along with a digital signature (Fig. 2). The NIST Digital Signature Standard (DSS) establishes three algorithms for signatures, RSA, Digital Signature Algorithm (DSA) and Elliptic Curve DSA (ECDSA) [28]. We choose DSA in our implementation, but this can replaced by either of the other algorithms with minimal performance impact. For hashing function, SHA-3 is chosen because it is the latest Cryptographic Hash Standard issued by NIST [14]. More specifically, the enrollment data is hashed using SHA3-256 and subsequently signed with the enroller's private key using an implementation of DSA with 3072-bit private key from the open-source Crypto++ library [1]. Details about performance are presented in Sec. 5.2.

4.3 Feature Verification

Verification compares the enrolled keypoints against the ROI of a new image in to order compute a similarity score. The integrity of enrolled keypoints is first verified by checking the digital signature. When a new image is captured for verification, its ROI is identified relative to the marker, and keypoints are extracted from the ROI. This mirrors the corresponding steps performed in feature enrollment, so we don't repeat their description here. The processing performed with the verification keypoints is as follows.

4.3.1 Feature Matching and RANSAC based Homography Computation

Two images of the same planar surface taken from different perspectives are related by a homography, which is a geometric model that maps feature positions in one image to the corresponding positions in the second image. Estimating the homography requires finding enrollment and verification keypoints that are similar and therefore likely to be representations of the same feature on the package surface. We find such points by performing nearest neighbor matching using OpenCV's FLANN (Fast Library for Approximate Nearest Neighbors) [40] matcher, and then evaluating quality of matches using a standard approach based on ratio of feature distances [33] as described here. For every keypoint k_i in ROI_{enroll}, we find its two closest (in feature distance) keypoints (k'_1 and k'_2) from ROI_{verify} and compute from their Euclidean distance in feature space a ratio score $r_i = \frac{\|k_i - k'_1\|_2}{\|k_i - k'_2\|_2}$. A low ratio indicates that keypoint k_i is significantly more similar to its best match k'_1 than to its second best match k'_2, which implies that k_i and k'_1 are likely to be corresponding points in the two images [33]. The 50 keypoint pairs with the lowest ratios (i.e., the best matches) are used as the basis for estimating a homography with the RANSAC algorithm. Increasing the number of matches will reduce the chance of RANSAC reaching consensus on an incorrect homography, but increases the expected number of random samples required to find consensus.

RANSAC (Random Sample Consensus) [15] is an algorithm to estimate a model from noisy data that contains both inliers and outliers. In our case, the computed model is the homography, and the data are the 50 selected keypoint pairs. RANSAC first samples four keypoint pairs from the set and calculates from them a homography matrix as in Eq. 1, where the 3x3 matrix is the homography, and P_e and P_v are the respective coordinates in enrollment and verification images of the keypoints. The quality of the homography model is then evaluated according to how many of the 50 keypoint pairs fit the model. Each pair that fits the homography model is considered an inlier. The process iterates to calculate and evaluate homographies from different sample points, and the homography with the highest number of inliers is returned as the best fit for the data.

$$P_v = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \times P_e \qquad (1)$$

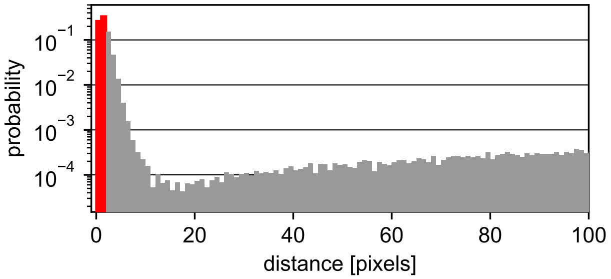
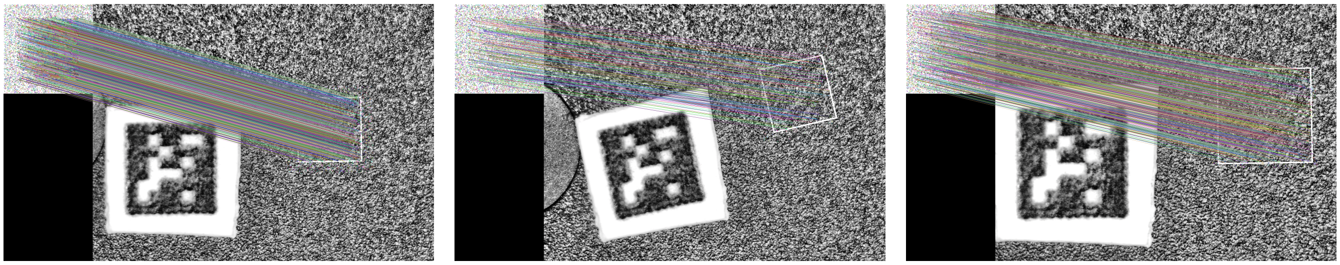


Figure 5: Pixel distances between enrolled keypoints and the verification keypoints that are their nearest feature-space neighbors. Correspondence of keypoint position is defined by homography. The spike at left comes from matched keypoints in the same relative positions, which are consistent with being from the same physical feature of the package. The points close enough to count as inliers are shaded red.

4.3.2 Projection and Scoring

Using the enrollment and verification keypoints, and the homography between them, we compute a score that indicates how many of the enrolled keypoints have good matches in the set of verification keypoints. An enrolled keypoint is considered to have a good match if there exists a verification keypoint that satisfies two conditions: (1) it is highly similar to the enrolled keypoint, and (2) it is at the position where the enrolled keypoint should be found in the verification image. The first condition is formalized as a requirement of being the nearest neighbor in feature space to the enrolled keypoint, and being at least 25% nearer than its second-closest neighbor (i.e. ratio score $r_i \leq 0.75$). The second condition is formalized as a requirement of being within 2 pixels of the location where the homography predicts the enrolled keypoint to be in the verification image. This ensures that matched features are not only similar, but also geometrically consistent with relative positions of the enrolled keypoints. Fig. 5 shows the pixel distance between the homography projection of an enrolled keypoint and the location of the verification keypoint that is its nearest neighbor in feature space. The data is collected from 100 different verification trials. The peak at left indicates that the nearest neighbor is often found within two pixels of the location predicted by the homography. These points are the inliers.

Fig. 6 shows examples of keypoint matching from verification. The matching succeeds even when the verification image is rotated and at a different scale from the orientation of the same chip at enrollment. Each line on the figure shows the correspondence between an enrolled keypoint and a matching keypoint found on the package during verification.



(a) Verification at nominal orientation

(b) Verification with rotation

(c) Verification at different scale

Figure 6: Three examples of matching between enrollment keypoints (square in upper left) and verification image of the same chip package instance, where the verification image differs in zoom and orientation. White square on chip package is the identified region of interest for verification. Each line corresponds to a keypoint match from enrollment to verification (Sec. 4.3.2).

5 System Evaluation

We evaluate the CounterFoil system using experiments on populations of two plastic dual in-line package (PDIP) chips. The first is an Alliance Memory AS6C6264-55PCN [3], which is a 64kb SRAM in a 28-pin PDIP (surface size 35.6mm × 15.2mm) that is rated for 0°C to 70°C temperature range. The second is a Microchip Technology 23LC1024 [39], which is a 1Mb SRAM in an 8-pin PDIP (9.2mm × 6.4mm) that is rated for −40°C to 85°C. Images are collected using two instances of two different camera models. The two ViTiny UM12 cameras [56] cost $390 each, have 5MP sensors, and computer-controlled focus through software. The two MustCam UM012C cameras cost $40 each, have 5MP sensors, and manual focus by turning a dial. Our collection of chips and cameras are shown in Fig 7.

In our evaluation we use 52 instances of chip model AS6C6264 and 40 instances of chip model 23LC1024. Chips packaged in the same mold are identified by the mold marking on the package. Our dataset has several chips packaged from the same mold: 5 pairs, 9 multiples in chip model AS6C6264 and 14 pairs in chip model 23LC1024. Each chip instance is enrolled to the database using one camera, and then verified using the other camera of the same model. Enrollment and verification is repeated 3 times for each chip, comprising a total of 528 images taken with ViTiny and MustCam.

5.1 Package Authentication

Package authentication is performed by matching verification image features with enrolled ones as described in Sec. 4. Fig. 8 shows in green the cumulative distribution function (CDF) of the number of inliers (matched keypoints) from the dataset of enrolled and verification chip images using our system. Fig. 8 also shows in red the CDF of inliers for mislabeled packages. In these cases, the program is modified to ignore the identity encoded on the label, and to fetch from the database the enrolled keypoints of another, randomly selected chip instance of the same model. 5,000 such comparisons are performed. This CDF represents what a counterfeiter might achieve by randomly swapping labels. We also consider the strongest adversary that has an exact duplicate of the mold that was used by the trusted packaging house to produce the enrolled chip, and he copies the label for the legitimate enrolled chip onto his counterfeits created from the same mold. The lines in blue show the number of inliers that the counterfeit would be able to achieve in this permissive setting. Even if the attacker has the same mold used to produce an enrolled chip, the counterfeits that can be created with the mold typically still have significantly fewer inliers than the enrolled chip.

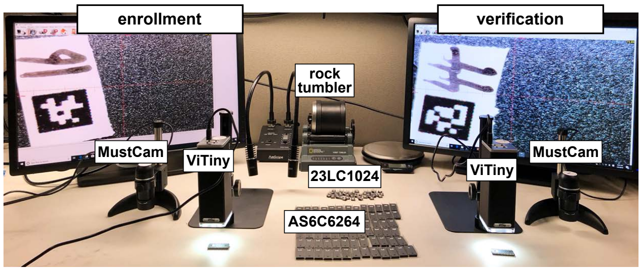
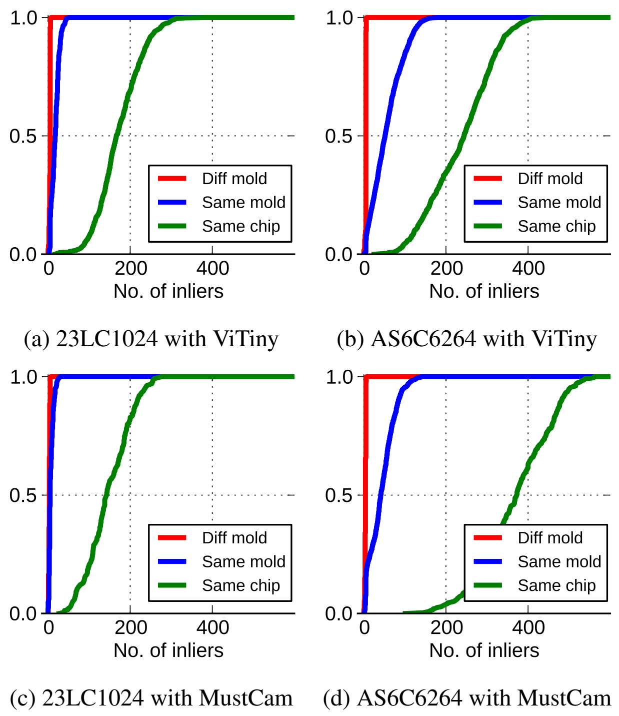


Figure 7: Experimental setup. Left side of workbench used for enrollment, right side used for verification. Separate camera are used for enrollment and verification. Middle of image shows the population of chips with labels affixed.

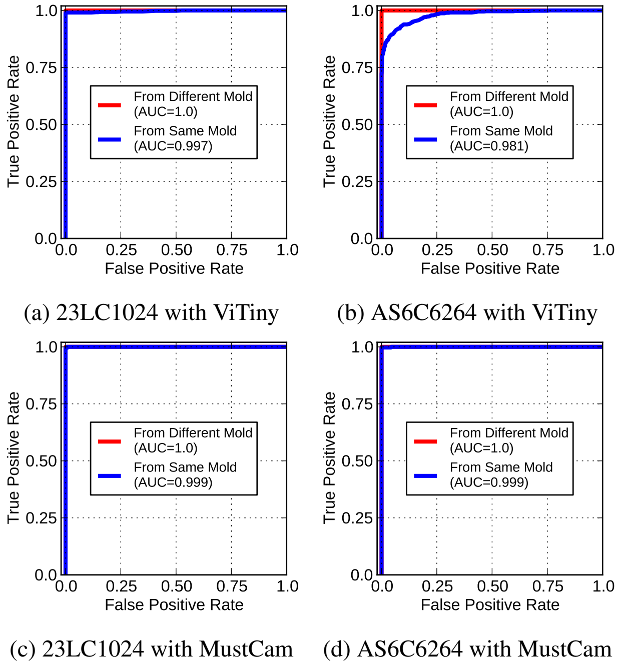
The verifier's decision to accept or reject a package is made according to whether the number of matched enrollment keypoints exceeds a threshold. A higher threshold is a more selective determination of authenticity. Higher thresholds can reduce both false positives (counterfeits accepted as authentic) and true positives (legitimate chips accepted as authentic). Receiver operating characteristic (ROC) curves are plots that show the achievable rates of true and false positives as the acceptance threshold is varied. A true positive always refers to a case where the enrolled and verified chip are the same instance with the same label, but we use two different notions of a false positive. The first case of false positive is



(a) 23LC1024 with ViTiny (b) AS6C6264 with ViTiny

(c) 23LC1024 with MustCam (d) AS6C6264 with MustCam

Figure 8: CDF of number of inliers using each model of camera.



(a) 23LC1024 with ViTiny (b) AS6C6264 with ViTiny

(c) 23LC1024 with MustCam (d) AS6C6264 with MustCam

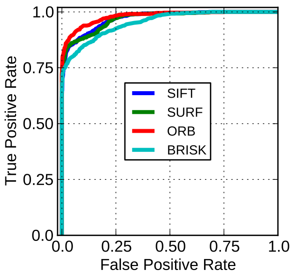
Figure 9: Receiver Operating Characteristic curves show ability to distinguish enrolled chips from other chips created from a different mold than the enrolled chip, or from the same mold that produced the enrolled chip.

a counterfeit chip with a label that was enrolled to a chip from a different mold. The second case of false positive is a counterfeit chip with a label used to enroll another chip from the same mold as itself. The first case corresponds to a typical unsophisticated counterfeiter, and the second is to provide an idea of what a determined and well-equipped attacker may be able to achieve. The ROC curves are shown in Fig. 9. For both models of chip and both models of camera, we are able to distinguish perfectly (100% true positives at 0% false positives) between a legitimate chip being verified and a counterfeit from a different mold. Even in the extreme case where the counterfeiter has the same mold (from the packaging house) used to create the enrolled chip, it is possible to detect the counterfeits while still keeping a high rate of true positives. The worst case is AS6C6264 with ViTiny camera (Fig. 9b), where it is still possible to accept 90% of legitimate chips while allowing only 10% of counterfeits created from the same mold. We will show later in the paper that this performance can be further improved by higher quality images. Note that the worst-case scenario of counterfeits from the same mold that produces legitimate chips demonstrates the effectiveness of CounterFoil, but our assumption of a trusted packaging house precludes an adversary having this capability.

CounterFoil is intended to be a scalable solution for provenance, so it is important to consider the possibility of collisions when enrolling fingerprints of many packages. Because the packaging house in possession of the molds is trusted, we focus on collisions that might occur in the ordinary scenario of a profit-seeking attacker that is using different molds to create counterfeit chips. A collision occurs when a verification fingerprint of package (A) is accepted as matching enrolled fingerprints from two different-mold packages (A) and (B). This collision is a true positive authentication of package (A), and a false positive authentication of package (A) against the enrolled fingerprint of (B). We are able to avoid false positives between different-mold chips in our limited dataset, so we use a simple model to estimate the false positive probability of a larger dataset. Enrolled fingerprints have an average of 3936 keypoints in a $2mm^2$ ROI, and we find empirically that each keypoint will become an inlier with probability 1.0E-3 when compared to a verification fingerprint from a different mold. Under the simplifying assumption that all keypoints have the same probability of being inliers, the number of inliers will follow a binomial distribution, and we can calculate the probability of inliers falsely exceeding the acceptance threshold. We choose for the model an acceptance threshold equal to the minimum number of inliers between same-chip comparisons, which is 48. The probability of having a false positive is then 5.6E-36, which is the estimated collision probability between two fingerprints from different molds. A collision probability of 5.6E-36 implies that the enrolled fingerprints have entropy of 117-bit random binary strings.

Table 1: Quantitative comparison of different feature-detecting methods. Plot at right shows the ROC plot from which the area-under-curve is computed. All four algorithms are configured to use 1,000 keypoints per mm^2 for this comparison.

Algorithm	Avg. Inliers		Area Under Curve	Run Time [s]
	Same Chip	Same Mold		
SIFT	570	178	0.971	0.215
SURF	470	100	0.970	0.211
ORB	236	56	0.980	0.064
BRISK	215	53	0.953	0.432



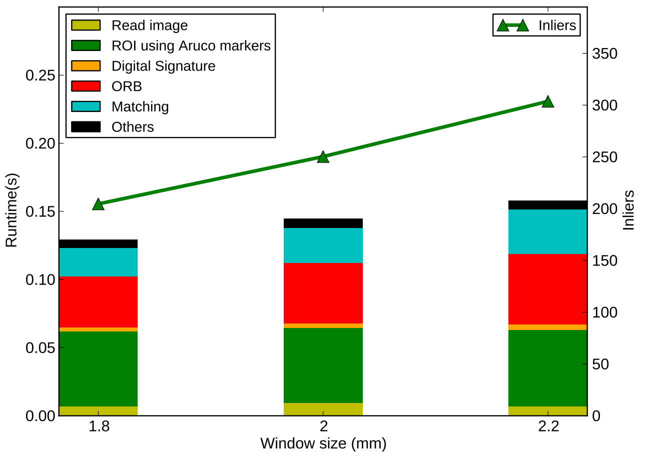


Figure 10: Runtime of verification procedure, broken down by processing task, for different sizes of ROI. Keypoint density is held constant at 1,000/mm^2. The increase in keypoints for the larger ROI results in a higher runtime, but also increases the number of matching points that are found. Runtime can be traded against accuracy by adjusting the ROI size.

5.2 Runtime

Verifying provenance of packages should not slow manufacturing (for enroller) or integration (for verifier). The verification process is more computationally intensive than enrollment, and certain target applications for verification may impose stringent latency requirements. For example, we envision that one application is integration with a pick-and-place machine, which removes chips from feeder reels and places them appropriately onto printed circuit board pads for reflow soldering. Single head pick-and-place machines from a leading manufacturer place between 1,800 and 5,000 parts per hour [37], which corresponds to handling each part for 720ms to 2s. Fig. 10 shows that package verification can be performed at production speed, as our system is able to authenticate each instance within 150 ms on an Intel Xeon CPU E5-2690. The runtime can be further reduced to meet even tighter latency requirements by enrolling a smaller number keypoints for each chip. Fig. 10 shows how runtime scales with the size of ROI at a constant keypoint density, and shows the breakdown of runtime by task. Enrolling a larger area of the chip surface increases the number of inliers and the total runtime. The next two subsections consider the runtime implications of algorithm choices.

5.2.1 Image Processing

Table 1 compares the runtime and authentication performance of four popular algorithms for feature extraction and matching. While all of the algorithms are suitable, we find ORB to perform best, and have thus chosen it for our work. In particular, the speedup of ORB comes largely from its compatibility of using locality-based hashing to identify near neighbors, without using the k-nearest neighbor search which is the most time consuming operation in the other algorithms.

5.2.2 Digital Signatures

We also evaluate the performance impact of using different digital signature algorithms such as DSA (3072-bit key) and ECDSA (256-bit key). For the one-time key generation step, ECDSA is significantly faster than DSA, with runtimes of 1.1ms and 2142ms respectively. More important is the runtime of the repeated steps of signing enrollment records and verifying signatures. Signing and verifying incur runtimes of 1.4ms and 1.6ms in DSA, and incur runtimes of 1ms and 2.6ms in ECDSA. Verification is the step with real-time constraints, so we use DSA over ECDSA, but the impact of this choice is minor because runtime is dominated by image processing. Further, signature verification can be done in parallel with feature verification and is not the performance bottleneck of COUNTERFOIL.

5.3 Practicality and Costs

The COUNTERFOIL methodology is compatible as an add-on to existing supply chains, and the cost at scale should be significantly less than one cent per chip. Chip verifiers can use the inexpensive camera models from our experiments, and perform processing on dedicated or shared computers. Given that verification would likely be performed at PCB assembly houses, the small cost of the camera would be insignificant, especially when amortized over a large number of boards being produced. The labels affixed to the chips cost $0.30 per sheet, and we print 1024 markers per sheet, for a per-unit cost of $0.0003 per label. The enrolled data for each chip is 1 MB, which at current hard-drive prices of $0.03 per GB corresponds to a per-unit cost of $0.00003 for storing the data. Affixing markers to each chip is currently a manual and time-consuming process. At scale we imagine that per-chip labels could be replaced by labels on part reels, or other ways of communicating a purported identity for

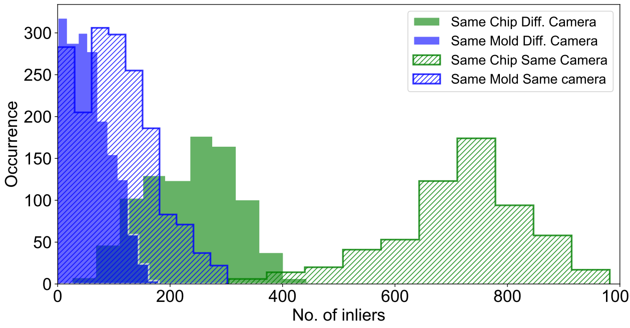


Figure 11: Histograms showing increase in number of inliers in AS6C6264 SRAM when same ViTiny cameras are used for both enrollment and verification.

the parts that would be used to access the signed enrollment records. In that case, the ROI would be identified based on image recognition of package surface instead of the markers. The low barriers to adoption of COUNTERFOIL are simply having a packaging house deploy the technology, and establishing keys for signing and verifying chips. Even if only a small fraction of purchasers would verify their chips using the available information, this should increase the risk of detection for distributors that traffic in possible counterfeits. The more significant barrier to adoption is perhaps the possibility that superficial cosmetic damage to parts could cause them to become untrusted, representing a monetary loss and a harm to branding.

Note that COUNTERFOIL is specifically targeted toward preventing inauthentic parts from being installed onto printed circuit boards of a system, and doing so without trusting distributors. The reliance on surface imaging makes the approach less compatible with authentication by intermediate distributors between packaging and deployment. Distributors that deal with parts in bulk will not ordinarily handle individual chips in a way that is conducive to surface imaging for COUNTERFOIL.

5.4 Camera Differences

Because enrollment and verification are performed using different camera instances, ability to match features may be impacted by differences in the lens, lighting, or the sensor array [34] of the cameras. To explore this further, we now evaluate how the matching performance changes in the unrealistic scenario of using the same ViTiny camera instance for both enrollment and verification of AS6C6264 chips, which was the most challenging authentication case in the prior experiments (see Fig 9b). Fig. 11 shows that using a consistent camera causes the number of inliers to increase, both in the case of same-chip comparisons and same-mold comparisons. The same-chip comparisons have a larger increase, and the overlap between the two distributions is reduced, implying capability for better authentication performance. This result reveals the presence of some detrimental camera variations that are being overcome in our realistic authentications that use different camera instances for verification and enrollment.

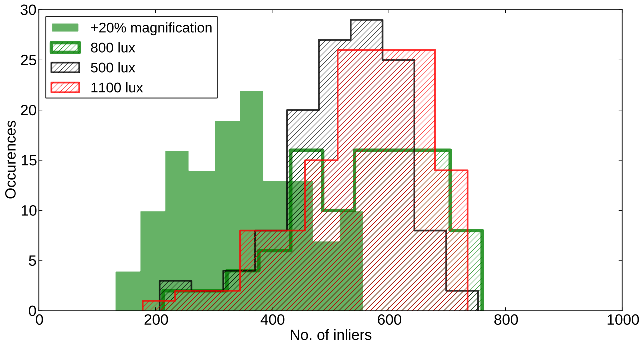


Figure 12: Histogram of inliers in AS6C6264 SRAM under two alternative lighting intensities (nominal is 800 lux) and one alternative zoom.

5.5 Varying Magnification and Lighting

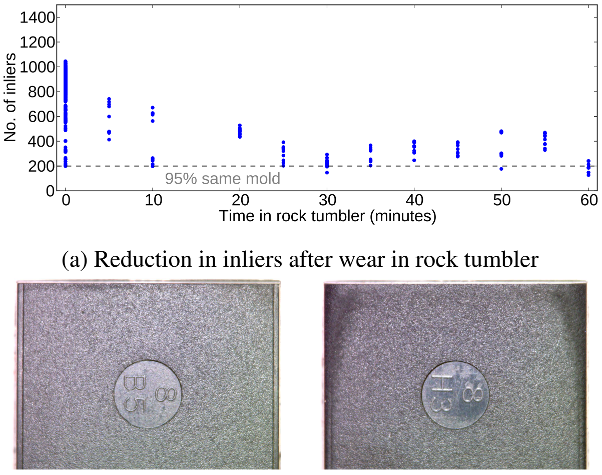
Fig. 12 shows results under different magnification and lighting conditions using the ViTiny camera with the AS6C6264 chips using a smaller dataset with 10 chip instances. The approach is largely unaffected by lighting changes, but changing the magnification from enrollment to verification has some impact on the number of inliers.

6 Further Investigation of Fingerprints

In this section we deviate from our standard system to investigate package fingerprint properties that cannot easily be evaluated within the overall system. In particular, for different reasons, experiments in this section define the ROI in a way that doesn't rely on affixed labels. Instead of defining the center of the ROI as being at position $< r, \theta >$ relative to the marker (see Fig. 4), the center of the ROI is here defined as a pixel in the center of the image. To ensure that the same area of the chip is always imaged, the chip is aligned carefully to the camera. Aside from lacking markers, the image processing performed is as described in Sec. 4.

6.1 Testing Resilience of Fingerprints

The fingerprints should be robust enough to withstand wear that occurs when IC packages are jostled and handled during distribution. We use various time durations in a hobbyist rock tumbler to impart controllable amounts of wear on chips. After enrollment, chips are placed alone in the rock tumbler with 45mL of water and 5g of 60-grit silicon carbide, which is the coarsest grit used in rock tumbling. The tumbler barrel



(a) Reduction in inliers after wear in rock tumbler

(b) Chip before tumbler

(c) Chip after 1 hour

Figure 13: Reduction in inliers for chip AS6C6264 after spending time in rock tumbler. Images of chip are included to give a sense of the amount of wear caused.

is washed out between experiments, and each trial uses new grit and clean water. After tumbling, the chip is removed, rinsed under a faucet, dried and imaged for verification. The prototype adhesive labels do not survive the rock tumbler, so the ROI in the images is instead found by careful alignment of the chip under the camera.

Fig. 13 shows the degradation in number of inliers for chips after different amounts of time in the tumbler. The plot shows a slow decrease in the number of inliers after tumbling with a few hundred inliers left after an hour in the tumbler. The dashed line on the plot shows the acceptance threshold that has a 95 percent probability of rejecting a different chip from the same mold. In other words, an attacker that has obtained the same mold and produced new chips from it will have only a 5% of exceeding this threshold and thereby succeeding in forgery. Even after significant wear, most authentication trials from the legitimate chip are able to exceed this value.

Figs. 13b and 13c show package surfaces before and after 1 hour in the tumbler. Note that these images are illustrative; they use a different magnification from the results in Fig. 13a and include the corners of the chip where the wear is most noticeable, instead of showing only the ROI where the wear is less apparent. We also tested the effect of temperature by heating the chips to 170°C for an hour in a thermal chamber, but saw no change in the number of inliers.

6.2 Testing Fingerprint Uniqueness

Any complex physical object has some combination of minute features that are unlike all other instances of the same object. Given that molded integrated circuit packages are heterogeneous mixtures of particles, they are certain to be unique in this trivial, physical, sense. However, for authentication the relevant question is whether there is a uniqueness that is observable and stable at the scale of our imaging. In studying uniqueness, we pay special attention to chips that are produced from the same mold. Fortunately, each chip bears a mold mark that is imprinted in a circle on the underside of the chip. The mold mark, as is visible in Fig. 13b, gives a code of one letter and two numbers. The marks are used for traceability within the packaging facility, so that problematic molds can be identified. Our experiments confirm that chips with the same mark are from the same mold, as they show a distinct similarity according to our analysis, and in fact a similar texture can be observed at high magnification.

6.2.1 Scoring under Controlled Alignment

Experiments that use imprecisely placed labels to define the ROI of each chip cannot definitively show whether package fingerprints are unique. Two packages that are identical would appear unique if their labels are placed in such a way that their ROIs are disjoint regions of the package surface. We again avoid relying on markers and perform experiments in which ROI is based on chip alignment underneath the camera. Fig. 14 shows the result. Different chip instances from the same mold do show similarity, but it is smaller than the similarity between two images of the same chip. In chip type AS6C6264, the highest score between any two images of different chips from the same mold is 277 inliers, whereas the lowest score between any two images of the same chip is 603 inliers; the means are 113 and 825 respectively. The clear difference in scores for same-mold and same-chip comparisons is significant, as it shows that the mold surface texture is not entirely responsible for the fingerprints. Even if an adversary were able to perfectly reproduce (or steal) the mold, they will be unable to create high quality forged packages with it.

6.2.2 PUF-like Evaluation using Pixel Intensity

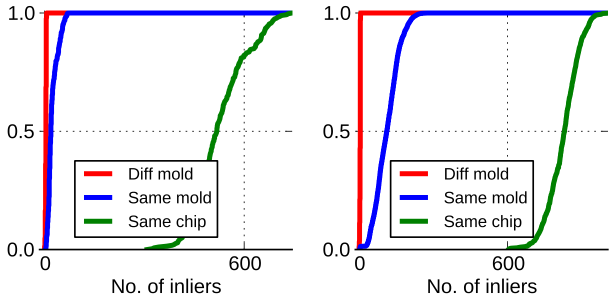
We also consider evaluating similarity of package fingerprints using a standard Physically Unclonable Function(PUF)-like scheme rather than the computer vision based techniques used in COUNTERFOIL. As standard PUF metrics [35,36] based on Hamming distance are not directly applicable in this setting, distance comparisons between enrollment and verification images are made by comparing the 8-bit pixel intensities of the two ROIs on a pixel-by-pixel basis, which is analogous to comparing responses from weak PUFs on a bit-by-bit basis.

The major challenge in making this comparison is that, unlike in digital PUFs, when comparing images there is no ground truth about which pixel in the verification image should be compared against which pixel in the enrollment image. Even if the package appears identical in the two images, the pixel-by-pixel comparison will only show the similarity if the two images have pixel-accurate alignment. Aside from requiring pixel-accurate alignment in the X and Y directions, rotation and scale variance additionally cannot be tolerated. Still, with some difficulty, we can partially overcome these challenges to make a pixel-by-pixel comparison. To make the comparison, we start from images taken using controlled alignment. A brute-force search is then performed to find the X and Y offset that best aligns the images, as seen in Fig. 15b. Only when the alignment is correct to within a few pixels does the similarity between the images become apparent. The need to perform brute force search for alignment increases runtime to 10s per comparison, which is hundreds of times slower than COUNTERFOIL, and still unable to handle any change to rotation or scale. The results from making hundreds of comparisons in this manner are shown in Fig. 15a. In some cases, presumably due to rotation or scale, the similarity between the same-chip images cannot be found using pixel-by-pixel comparisons. This result confirms that the package features can with some difficulty be observed in a PUF like way, but also shows that pixel-by-pixel comparisons are not well-suited to this task relative to the computer vision approach.

6.2.3 PUF-like Evaluation using Feature Distance

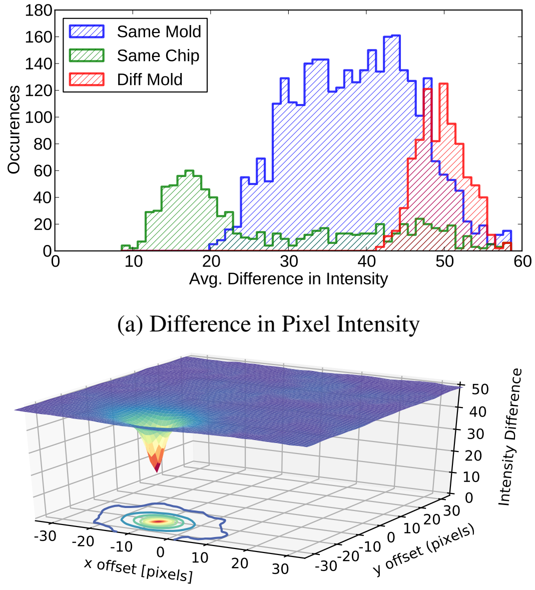
In COUNTERFOIL, the number of matches that we compute as inliers is based on both feature similarity, and the geometric relationship of the features on the package surface, as matched keypoints from enrollment and verification must be related by a homography. One might also consider evaluating similarity of the features in corresponding positions of two chip packages, similar to Hamming Distance between corresponding bits in a PUF circuit. In this case, the computer vision approach is being used to align the enrollment and verification keypoints, but after alignment is decided the corresponding features are scored according to their similarity in feature space instead of their pixel intensity.

Fig. 16 shows the average distance, in feature space, between features having positional correspondence defined by computed homography. In a highly controlled setting of careful alignment, lighting and single camera, the same package can be distinguished from packages created from the same mold, as shown by the separation between the feature distances in Fig. 16a. However, in the general setting which contains typical image quality variations, the same chip distribution is shifted to the right leading to a slight overlap with the same mold distribution as shown in Fig. 16a. An absolute feature distance threshold to distinguish between chips from same mold is therefore not robust to image quality variations. COUNTERFOIL aims to avoid this limitation by using feature similarity ranking (nearest neighbors) instead of an absolute distance threshold.



(a) 23LC1024 with ViTiny (b) AS6C6264 with ViTiny

Figure 14: Inlier CDFs for SRAMs under controlled alignment.



(a) Difference in Pixel Intensity

(b) Alignment

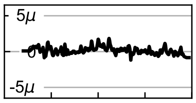
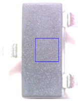
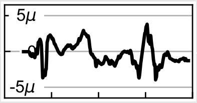
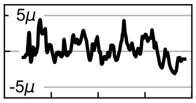
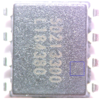
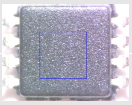
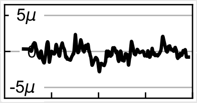
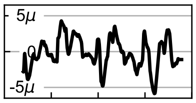
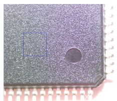
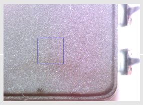
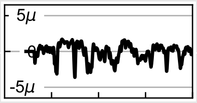
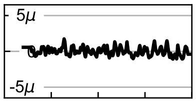
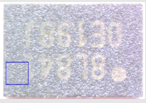
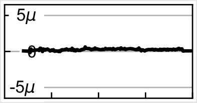
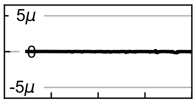
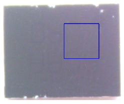
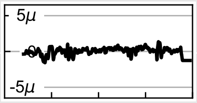
Figure 15: PUF-like evaluation on raw pixel intensity data.

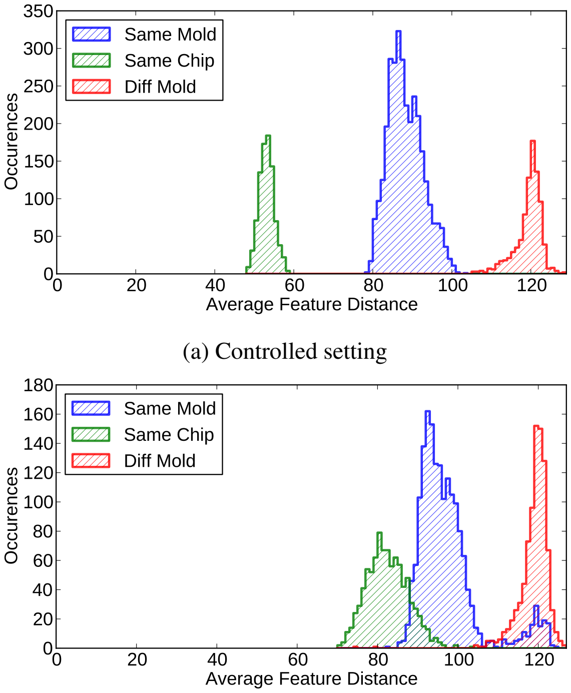
6.3 Additional Package Types

To further validate package surface fingerprints, we conduct experiments with 10 additional circuit package types. As before, one ViTiny camera is used for enrollment, and a second for verification. We use 5 instances of each chip, and from each instance collect 5 enrollment and 5 verification images. Note that, among the molded packages in this secondary population, none appear to be from the same mold.

Table 2 summarizes the results of the experiment. Because many of the packages are quite small, and we want to use an unmarked area of the package surface as the fingerprint, in some cases the enrolled area of the surface is smaller than $2mm^2$. ROI is identified by manual chip alignment under the camera, as many of the packages are impractically small for

Table 2: Evaluation of package surface fingerprints across a range of package types. Contrast between number of inliers in same chip comparisons and different chip comparisons is an indication of suitability to COUNTERFOIL.

Surface Map	Example Image	Chip Name	Package	Same Chip Inliers μ (σ)	Diff. Chip Inliers μ (σ)	Area (mm^2)	Example Image	Surface Map
		W25Q80EWUXIETR	23-SOT	38.6 (13.2)	3.0 (1.5)	0.454		
		TSV524IQ4T	16-QFN	42.9 (8.4)	4.1 (1.7)	0.315		
		MX25V4006EM1I-13G	8-SOIC	58.8 (10.2)	3.8 (1.5)	0.454		
		24LC32A-I/MS	8-MSOP	344.3 (44.8)	4.0 (1.3)	2		
		CY7C1353G-100AXC	100-TQFP	280.8 (40.1)	4.7 (1.0)	2		
		ADG419TQ	14-CDIP	358.4 (73.2)	3.9 (1.3)	2		
		ADP125ACPZ-R7	8-LFCSP	18.3 (7.2)	3.2 (1.5)	0.315		
		W25Q80EWUXIE TR	8-USON	12.3 (5.9)	2.1 (1.3)	0.201		
		FAN53540UCX	20-WLCSP	3.2 (2.9)	1.8 (1.4)	0.315		
		2N3440	TO-39	0 (0)	0 (0)	2		



(a) Controlled setting

(b) Uncontrolled setting

Figure 16: Average distance in feature space for same-position keypoint pairs.

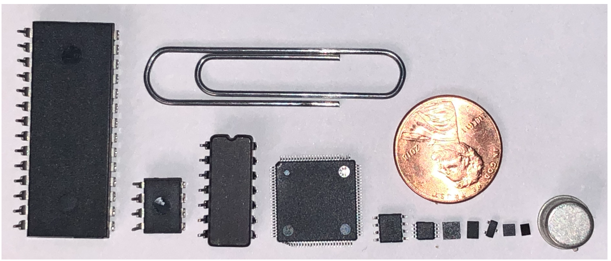


Figure 17: Tested package types include plastic, ceramic, metal and waver level packages. Paperclip and U.S. one-cent coin are shown for scale.

the crude adhesive markers used in our prototype demonstration. The table gives for each chip an example image with the ROI marked by a square. To give a sense of the surface structure of each package model, we plot within the table the deviation from nominal surface height along an arbitrary 0.9mm trace of the surface; this data is collected with the same Zygo Nexview 3D optical surface profiler used to generate Fig. 3a.

The significant distance between the average number of inliers for same chip and different chip comparisons implies that it may be possible to authenticate most of the plastic packages by their fingerprints, although further experiments would be needed to give confidence. Interestingly, based on this preliminary data, the ceramic package (14-CDIP) also appears to have identifying features. Two packages that are notably unsuitable for the style of package fingerprinting used in this paper are the final two entries in the table – the TO-39 metal can package and 20-WLCSP wafer-level package. In these two cases, the reflective surfaces cause very few keypoints to be extracted from the image, and the extracted keypoints do not match well between enrollment and verification.

7 Conclusion

In this paper we have presented CounterFoil, a system that verifies provenance by extracting unique fingerprints from surface features of integrated circuit packages imaged using inexpensive cameras. The work is a low-cost strategy that can help to address the significant problem of counterfeit integrated circuits which results in billions of dollars of losses each year. Our approach enrolls unique features of each chip after packaging, and requires no chain-of-custody through distribution. During verification features are matched against cryptographically signed enrollment records. We've demonstrated the approach to work on a large population of two different chips, have used different models of low-cost microscope cameras, and have evaluated resiliency of fingerprints. Crucially, we've shown that even an adversary possessing an exact duplicate of the mold used to produce a chip's package will not easily be able to create a high-quality counterfeit of the chip.

Acknowledgments

We thank the anonymous reviewers and our shepherd, Yossi Oren, for their insightful comments on earlier versions of this work. This work was supported in part by NSF grant CNS-1749845.

Availability

The code and dataset of images used in this paper are available at https://github.com/danholcomb/supply-chain-security

References

[1] Crypto++ Library 8.1.0, Feb 2019. https://www.cryptopp.com/.

[2] N. E. C. Akkaya, B. Erbagci, and K. Mai. Secure chip odometers using intentional controlled aging. In *2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 111–117, April 2018.

[3] Alliance Memory Inc. AS6C6264: 8k x 8bit Low Power CMOS SRAM, 2017. https://www.alliancememory.com/wp-content/uploads/pdf/Alliance%20Memory_64K_AS6C6264v2.0July2017.pdf.

[4] Navid Asadizanjani, Nathan Dunn, Sachin Gattigowda, Mark Tehranipoor, and Domenic Forte. A database for counterfeit electronics and automatic defect detection based on image processing and machine learning. *ISTFA, Nov*, 2016.

[5] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: speeded up robust features. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Computer Vision – ECCV 2006*, pages 404–417, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[6] Richard C Benson, Dawnielle Farrar, and Joseph A Miragliotta. Polymer adhesives and encapsulants for microelectronics applications. *Johns Hopkins APL Technical Digest*, 28(1):58, 2008.

[7] T. D. Bergman, C. P. Manager, and K. T. Liszewski. Battelle barricade: A nondestructive electronic component authentication and counterfeit detection technology. In *2016 IEEE Symposium on Technologies for Homeland Security (HST)*, pages 1–6, May 2016.

[8] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[9] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 93–118. Springer, 2001.

[10] Christopher Henderson. Transfer Molding, 9 2012. In InfoTracks Semitracks Monthly Newsletter; Available: http://www.semitracks.com/newsletters/september/2012-september-newsletter.pdf.

[11] W. Clarkson, T. Weyrich, A. Finkelstein, N. Heninger, J. A. Halderman, and E. W. Felten. Fingerprinting blank paper using commodity scanners. In *2009 30th IEEE Symposium on Security and Privacy*, pages 301–314, May 2009.

[12] W. E. Cobb, E. D. Laspe, R. O. Baldwin, M. A. Temple, and Y. C. Kim. Intrinsic physical-layer authentication of integrated circuits. *IEEE Transactions on Information Forensics and Security*, 7(1):14–24, Feb 2012.

[13] Gerald DeJean and Darko Kirovski. RF-DNA: Radio-Frequency Certificates of Authenticity. In Pascal Paillier and Ingrid Verbauwhede, editors, *Proceedings of the 9th International Conference on Cryptographic Hardware and Embedded Systems - CHES 2007*, pages 346–363, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[14] Morris J Dworkin. SHA-3 standard: Permutation-based hash and extendable-output functions. Technical report, 2015.

[15] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[16] Francis Galton. *Fingerprint directories*. Macmillan and Company, 1895.

[17] S. Garrido-Jurado, R. Muñoz Salinas, F.J. Madrid-Cuevas, and M.J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recogn.*, 47(6):2280–2292, June 2014.

[18] B Gassend, D Clarke, and M Van Dijk. Silicon physical random functions. In *Proceedings of the IEEE Computer and Communications Society*, 2002.

[19] J Guajardo, S Kumar, GJ Schrijen, and P Tuyls. FPGA intrinsic PUFs and their use for IP protection. *Cryptographic Hardware and Embedded Systems*, 2007.

[20] Ujjwal Guin, Ke Huang, Daniel DiMase, John M Carulli, Mohammad Tehranipoor, and Yiorgos Makris. Counterfeit integrated circuits: a rising threat in the global semiconductor supply chain. *Proceedings of the IEEE*, 102(8):1207–1228, 2014.

[21] Ujjwal Guin, Xuehui Zhang, Domenic Forte, and Mohammad Tehranipoor. Low-cost on-chip structures for combating die and IC recycling. In *Proceedings of the 51st Annual Design Automation Conference*, DAC '14, pages 87:1–87:6, New York, NY, USA, 2014. ACM.

[22] Ghaith Hammouri, Aykutlu Dana, and Berk Sunar. CDs have fingerprints too. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 348–362. Springer, 2009.

[23] James A Hayward and Janice Meraglia. DNA marking and authentication: A unique, secure anti-counterfeiting program for the electronics industry. In *International Symposium on Microelectronics*, volume 2011, pages 000107–000112. International Microelectronics Assembly and Packaging Society, 2011.

[24] Daniel E. Holcomb, Wayne P. Burleson, and Kevin Fu. Power-up SRAM state as an identifying fingerprint and source of true random numbers. *IEEE Transactions on Computers*, 58(9):1198–1210, September 2009.

[25] Wan-Chiech Huang, Chao-Ming Hsu, and Cheng-Fu Yang. Recycling and refurbishing of epoxy packaging mold ports and plungers. *Inventions*, 1(2):11, 2016.

[26] IHS Technology. Top 5 most counterfeited parts represent a $169 billion potential challenge for global semiconductor market, 2012. Available: http://www.isuppli.com/Semiconductor-Value-Chain/News/pages/Top-5-Most-Counterfeited-Parts-Represent-a-\protect\T1\textdollar169-Billion-Potential-Challenge-for-Global-Semiconductor-Market.aspx.

[27] N. Kae-Nune and S. Pesseguier. Qualification and testing process to implement anti-counterfeiting technologies into IC packages. In *2013 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 1131–1136, March 2013.

[28] C Kerry and P Gallagher. FIPS PUB 186-4: digital signature standard (DSS). *FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION. National Institute of Standards und Technology*, 2013.

[29] Eric Koziel, Kate Thurmer, Lauren Milechin, Peter Grossmann, Michael Vai, Roger Khazan, Keith Bergevin, and Philip Comer. Side channel authenticity discriminant analysis for device class identification. In *Government Microciruit Applications & Critical Technology Conference*, 2016.

[30] Serge Leef. Supply Chain Hardware Integrity for Electronics Defense (SHIELD), 2018. Available: https://csrc.nist.gov/CSRC/media/Projects/cyber-supply-chain-risk-management/documents/SSCA/Winter_2018/TuePM2.1-SHIELD.pdf.

[31] S. Leutenegger, M. Chli, and R. Y. Siegwart. BRISK: binary robust invariant scalable keypoints. In *2011 International Conference on Computer Vision*, pages 2548–2555, Nov 2011.

[32] Zhengxiong Li, Aditya Singh Rathore, Chen Song, Sheng Wei, Yanzhi Wang, and Wenyao Xu. Printracker: Fingerprinting 3d printers using commodity scanners. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, pages 1306–1323, New York, NY, USA, 2018. ACM.

[33] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov 2004.

[34] Jan Lukáš, Jessica Fridrich, and Miroslav Goljan. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, 2006.

[35] Roel Maes and Ingrid Verbauwhede. Physically unclonable functions: a study on the state of the art and future research directions. In *in Towards Hardware-Intrinsic Security, Security and Cryptology*, 2010.

[36] Abhranil Maiti, Vikash Gunreddy, and Patrick Schaumont. A systematic method to evaluate and compare the performance of physical unclonable functions. cryptology eprint archive, report 2011/657, 2011.

[37] Manncorp. SMT Pick and Place Machines, 2019. https://www.manncorp.com/component-placement-and-handling.

[38] Michael L. Jones. DNA marking technology improves quality through fraud prevention, Sept 2016. Available: http://www.dla.mil/AboutDLA/News/NewsArticleView/Article/958928/dna-marking-technology-improves-quality-through-fraud-prevention/.

[39] Microchip Technology Inc. 23A1024/23LC1024: 1Mbit SPI Serial SRAM with SDI and SQI Interface, 2015. http://ww1.microchip.com/downloads/en/DeviceDoc/20005142C.pdf.

[40] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In Alpesh Ranchordas and Helder Araújo, editors, *VISAPP (1)*, pages 331–340. INSTICC Press, 2009.

[41] NASA JPL/OSMS Assurance Technology Program Office. Electric, Electronic and Electromechanical Parts Bulletin newsletter, 2011. available at https://nepp.nasa.gov/files/20647/2011%20EEE%20Parts%20Bulletin%20MayJune11%206_22_11.pdf.

[42] National Research Council. *Counterfeit deterrent features for the next-generation currency design*, volume 472. National Academies Press, 1993.

[43] M. Pecht and S. Tiku. Bogus: electronic manufacturing and consumers confront a rising tide of counterfeit electronics. *IEEE Spectrum*, 43(5):37–46, May 2006.

[44] Michael O Rabin. How to exchange secrets with oblivious transfer. *IACR Cryptology ePrint Archive*, 2005:187, 2005.

[45] Report of the Committee on Armed Services United States Senate; 112th congress. INQUIRY INTO COUNTERFEIT ELECTRONIC PARTS IN THE DEPARTMENT OF DEFENSE SUPPLY CHAIN, 2012. Available: https://www.armed-services.senate.gov/imo/media/doc/Counterfeit-Electronic-Parts.pdf.

[46] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: an efficient alternative to SIFT or SURF. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2564–2571, Washington, DC, USA, 2011. IEEE Computer Society.

[47] SAE International. Counterfeit Electronic Parts; Avoidance, Detection, Mitigation, and Disposition, Users, 2012. Revised 2016-09-12.

[48] Senate Armed Services Committee Hearing on Counterfeit Electronic Parts in the Defense Supply Chain. TESTIMONY OF RALPH L. DENINO Vice President Corporate Procurement L-3 Communications Corporation, Nov 2011. Available: https://www.armed-services.senate.gov/imo/media/doc/DeNino%2011-08-11.pdf.

[49] Ashlesh Sharma, Vidyuth Srinivasan, Vishal Kanchan, and Lakshminarayanan Subramanian. The fake vs real goods problem: Microscopy and machine learning to the rescue. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, pages 2011–2019, New York, NY, USA, 2017. ACM.

[50] J. P. Skudlarek, T. Katsioulas, and M. Chen. A platform solution for secure supply-chain and chip life-cycle management. *Computer*, 49(8):28–34, Aug 2016.

[51] SparkFun Electronics Blog. Fake ICs Identified, July 2010. Available: https://www.sparkfun.com/news/395.

[52] KW Tong, CK Kwong, and KW Ip. Optimization of process conditions for the transfer molding of electronic packages. *Journal of Materials Processing Technology*, 138(1):361–365, 2003.

[53] Rao R Tummala. Fundamentals of microsystems packaging. 2001.

[54] Pim Tuyls and Lejla Batina. RFID-Tags for Anticounterfeiting. In David Pointcheval, editor, *Topics in Cryptology – CT-RSA 2006*, pages 115–131, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[55] Pim Tuyls, Geert-Jan Schrijen, Boris Škorić, Jan van Geloven, Nynke Verhaegh, and Rob Wolters. Read-proof hardware from protective coatings. In *Cryptographic Hardware and Embedded Systems - CHES*, pages 369–383, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[56] ViTiny USA. ViTiny UM12 Long Working Distance 5MP USB Digital Microscope, 2018. http://www.vitiny-usa.com/vitiny-um12.html.

[57] Yu Zheng, Abhishek Basak, and Swarup Bhunia. CACI: Dynamic current analysis towards robust recycled chip identification. In *Proceedings of the 51st Annual Design Automation Conference*, DAC '14, pages 88:1–88:6, New York, NY, USA, 2014. ACM.

[58] Zygo Corporation. Nexview 3D Optical Surface Profiler. https://www.zygo.com/?/met/profilers/nexview/&utm_source=zygo&utm_medium=QualityMag&utm_content=NexviewPage&utm_campaign=PrintAd.