# On the Limits of Computational Fuzzy Extractors

Kenji Yasunaga* Kosuke Yuzawa†

October 28, 2014

**Abstract**

Fuller et al. (Asiacrypt 2013) studied on computational fuzzy extractors, and showed, as a negative result, that the existence of a computational "secure sketch" implies the existence of an information-theoretically secure sketch with slightly weaker parameters. In this work, we show a similar negative result such that, under some computational assumption, the existence of a computational fuzzy extractor also implies the existence of an information-theoretic fuzzy extractor with slightly weaker parameters. The assumption is that the generation procedure of the fuzzy extractor can be efficiently invertible. This result implies that to circumvent the limitations of information-theoretic fuzzy extractors, we need to employ computational fuzzy extractors in which the generation procedure cannot be efficiently invertible.

## 1 Introduction

Cryptographic primitives generally require uniformly random strings. A *fuzzy extractor* is a primitive proposed in [3] that can reliably derive uniformly random keys from noisy sources, such as biometric data (fingerprint, iris, facial image, etc.) and long pass-phrases.

Formally, Dodis et al. [3] defined fuzzy extractors to be a pair of procedures $(\mathsf{Gen}, \mathsf{Rep})$. The key generation procedure $\mathsf{Gen}$ receives a sample $w$ from a noisy source $W$ with some entropy, and outputs a uniformly random key $r$ and a helper string $p$. After that, the reproduction procedure $\mathsf{Rep}$ can be used to derive the same key $r$ from the helper string $p$ and a sample $w'$ that is close to the original sample $w$. Notably, this framework does not need secret keys other than $w$. The derived key $r$ is close to uniform even if the helper string $p$ was given. See [4, 1] for surveys of results related to fuzzy extractors.

Dodis et al. [3] introduced the notion of *secure sketch*, which, on input $w$, produces an information that enables the recovery of $w$ from any close value $w'$ and does not reveal much information about $w$. Then, they show that a combination of secure sketches and strong extractors gives fuzzy extractors.

Fuzzy extractors are defined as *information-theoretic* primitives. Several limitations regarding parameters in fuzzy extractors are also studied in [3]. The *entropy loss* is the difference between the entropy of $w$ and the length of the extracted key $k$. In the setting of information-theoretic security, the entropy loss is known to be inevitable [6].

*Institute of Science and Engineering, Kanazawa University. Kakuma-machi, Kanazawa, 920–1192, Japan. yasunaga@se.kanazawa-u.ac.jp

†Graduate School of Natural Science and Technology, Kanazawa University. Kakuma-machi, Kanazawa, 920-1192, Japan. makku107@stu.kanazawa-u.ac.jp

Fuller et al. [5] consider the *computational security* of fuzzy extractors to circumvent the limitations of information-theoretic fuzzy extractors. They gave both negative and positive results. On one hand, they show that secure sketches with computational security need to be subject to lower bounds from coding theory. In particular, they show that the existence of a computational secure sketch implies the existence of an information-theoretic secure sketch with slightly weaker parameter. On the other hand, they present a direct construction of a computational fuzzy extractor based on the hardness of learning with errors (LWE) problem.

In this work, we show that, assuming that the generation procedure $\mathsf{Gen}$ can be efficiently invertible, computational fuzzy extractors also need to be subject to lower bounds from coding theory. Specifically, we show that if $w$ can be efficiently computable from the pair $(r, p)$ that can be generated by $\mathsf{Gen}(w)$, then the existence of a computational fuzzy extractor implies the existence of an information-theoretic fuzzy extractor with slightly weaker parameters. This negative result implies that in order to circumvent the limitation of the entropy loss of information-theoretic fuzzy extractors, we need to employ computational fuzzy extractors in which the generation procedure cannot be efficiently invertible. Indeed, there are extractors for structured sources that can be efficiently invertible [2].

### Comparison to the Results of Fuller et al. [5]

As noted in [5], when assuming that the generation procedure $\mathsf{Gen}$ can be efficiently invertible, their negative results for computational secure sketches can also be applied to computational fuzzy extractors. Here we describe this fact in more detail, and compare it with our result. Let $(\mathsf{Gen}, \mathsf{Rep})$ be a computational fuzzy extractor. Assume that there is an efficient algorithm $\mathsf{InvGen}$ that, given $(r, p)$, output $w$, where $(r, p)$ was generated by $\mathsf{Gen}(w)$. Then, one can construct a computational secure sketch $(\mathsf{SS}, \mathsf{Rec})$ (see Definition 3 for the definition of secure sketches) by defining $\mathsf{SS}(w) = \{(r, p) \leftarrow \mathsf{Gen}(w); \text{Output } p\}$ and $\mathsf{Rec}(w', \mathsf{SS}(w)) = \{r \leftarrow \mathsf{Rep}(w', p); w \leftarrow \mathsf{InvGen}(r, p); \text{Output } w\}$. Thus, by the negative results of [5], this implies the existence of secure sketch and fuzzy extractor with information-theoretic security. However, the above observation can be applied only if $\mathsf{InvGen}(r, p)$ outputs the same $w$ from which $(r, p)$ was actually generated. In general, $\mathsf{Gen}$ is not injective. Namely, there could exist different $w_1$ and $w_2$ such that the outputs of $\mathsf{Gen}(w_1)$ and $\mathsf{Gen}(w_2)$ are the same. (In particular, this happens when $r$ is relatively short.) In such a case, at least one of $w_1$ and $w_2$ cannot be recovered by $\mathsf{InvGen}$, and thus it seems difficult to use $\mathsf{InvGen}$ for constructing secure sketches. In contrast, we give our negative result for computational fuzzy extractors even when $\mathsf{InvGen}$ is not injective. In this sense, our result can be seen as a generalization of the negative results of [5].

## 2 Preliminaries

Let $X$ and $Y$ be random variables over some alphabet $Z$. The *min-entropy* of $X$ is $\mathrm{H}_\infty(X) = -\log(\max_x \Pr[X = x])$. The *average min-entropy* of $X$ given $Y$ is $\tilde{\mathrm{H}}_\infty(X|Y) = -\log(\mathbb{E}_{y \in Z} \max_{x \in Z} \Pr[X = x|Y = y])$. The *statistical distance* between $X$ and $Y$ is $\Delta(X, Y) = \frac{1}{2} \sum_{z \in Z} |\Pr[X = z] - \Pr[Y = z]|$. If $\Delta(X, Y) \leq \epsilon$, we say $X$ and $Y$ are *$\epsilon$-close*. We denote by $U_\ell$ the uniformly distributed random variable on $\{0, 1\}^\ell$. For $s \in \mathbb{N}$, the *computational distance* between $X$ and $Y$ is $\Delta^s(X, Y) = \max_{D \in \mathcal{C}_s} |\mathbb{E}[D(X)] - \mathbb{E}[D(Y)]|$, where $C_s$ is the set of randomized circuits of size at most $s$ that output 0 or 1. A metric space is a set $\mathcal{M}$ with a distance function

$\mathsf{dis} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}^+ = [0, \infty)$. For the Hamming metric over $Z^n$, $\mathsf{dis}(x, y)$ is the number of positions in which $x$ and $y$ differ. For a probabilistic experiment $E$ and a predicate $P$, we denote by $\Pr[E : P]$ the probability that the predicate $P$ is true after the event $E$ occurred.

We give definitions of fuzzy extractor, computational fuzzy extractor, secure sketch, and strong extractor.

**Definition 1** (Fuzzy Extractor)**.** *An* $(\mathcal{M}, m, \ell, t, \epsilon)$-fuzzy extractor *with error* $\delta$ *is a pair of randomized procedures* $(\mathsf{Gen}, \mathsf{Rep})$ *with the following properties:*

- *The generation procedure* $\mathsf{Gen}$ *on input* $w \in \mathcal{M}$ *outputs an extracted string* $r \in \{0, 1\}^\ell$ *and a helper string* $p \in \{0, 1\}^*$*.*

- *The reproduction procedure* $\mathsf{Rep}$ *takes* $w' \in \mathcal{M}$ *and* $p \in \{0, 1\}^*$ *as inputs. The* correctness *property guarantees that for any* $w, w' \in \mathcal{M}$ *with* $\mathsf{dis}(w, w') \le t$*, if* $(R, P) \leftarrow \mathsf{Gen}(w)$*, then* $\mathsf{Rep}(w', P) = R$ *with probability at least* $1 - \delta$*, where the probability is taken over the coins of* $\mathsf{Gen}$ *and* $\mathsf{Rep}$*. If* $\mathsf{dis}(w, w') > t$*, no guarantee is provided about the output of* $\mathsf{Rep}$*.*

- *The* security *property guarantees that for any distribution* $W$ *on* $\mathcal{M}$ *of min-entropy* $m$*, if* $(R, P) \leftarrow \mathsf{Gen}(W)$*, then* $\Delta((R, P), (U_\ell, P)) \le \epsilon$*.*

**Definition 2** (Computational Fuzzy Extractor)**.** *An* $(\mathcal{M}, m, \ell, t, s, \epsilon)$-computational fuzzy extractor *with error* $\delta$ *is a pair of randomized procedures* $(\mathsf{Gen}, \mathsf{Rep})$ *that is an* $(\mathcal{M}, m, \ell, t, \epsilon)$*-fuzzy extractor with error* $\delta$ *in which the security property is replaced by the following one:*

- *For any distribution* $W$ *on* $\mathcal{M}$ *of min-entropy* $m$*, if* $(R, P) \leftarrow \mathsf{Gen}(W)$*, then* $\Delta^s((R, P), (U_\ell, P)) \le \epsilon$*.*

**Definition 3** (Secure Sketch)**.** *An* $(\mathcal{M}, m, \tilde{m}, t)$-secure sketch *with error* $\delta$ *is a pair of randomized procedures* $(\mathsf{SS}, \mathsf{Rec})$ *with the following properties:*

- *The sketching procedure* $\mathsf{SS}$ *on input* $w \in \mathcal{M}$ *outputs a string* $s \in \{0, 1\}^*$*.*

- *The recovery procedure* $\mathsf{Rec}$ *takes* $w' \in \mathcal{M}$ *and* $s \in \{0, 1\}^*$ *as inputs. The* correctness *property guarantees that for any* $w, w' \in \mathcal{M}$ *with* $\mathsf{dis}(w, w') \le t$*,* $\Pr[\mathsf{Rec}(w', \mathsf{SS}(s)) = w] \ge 1 - \delta$ *where the probability is taken over the coins of* $\mathsf{SS}$ *and* $\mathsf{Rec}$*. If* $\mathsf{dis}(w, w') > t$*, no guarantee is provided about the output of* $\mathsf{Rec}$*.*

- *The* security *property guarantees that for any distribution* $W$ *on* $\mathcal{M}$ *of min-entropy* $m$*,* $\tilde{\mathrm{H}}_\infty(W|\mathsf{SS}(W)) \ge \tilde{m}$*.*

**Definition 4.** *We say that* $\mathsf{Ext} : \{0, 1\}^n \to \{0, 1\}^\ell$ *is an* $(n, m, \ell, \epsilon)$-strong extractor *if for any* $W$ *on* $\{0, 1\}^n$ *of min-entropy* $m$*,* $\Delta((\mathsf{Ext}(W; X), X), (U_\ell, X)) \le \epsilon$*, where* $X$ *is the uniform distribution on* $\{0, 1\}^r$*.*

# 3 Main Results

In this section, we show that the existence of a computational fuzzy extractor implies the existence of an information-theoretic fuzzy extractor with slightly weaker parameters. For this result, we need a computational assumption that the generation procedure of a fuzzy extractor can be efficiently invertible.

We give a formal definition of invertibility of the generation procedure.

**Definition 5.** *Let* $(\mathsf{Gen}, \mathsf{Rep})$ *be a fuzzy extractor for a metric space* $\mathcal{M}$*. We say* $\mathsf{Gen}$ *is* $(s, \eta)$-invertible *if there exists a deterministic circuit* $\mathsf{InvGen}$ *of size at most* $s$ *such that*

$$\Pr\left[w' \leftarrow \mathsf{InvGen}(R', p) : \exists\, r_G \in \{0,1\}^* \text{ s.t. } \mathsf{Gen}(w'; r_G) = (R', p)\right] \geq 1 - \eta$$

*for any* $p$ *that can be generated as* $(r, p) \leftarrow \mathsf{Gen}(w)$ *for* $w \in \mathcal{M}$*, where* $R' = U_\ell$*. We say* $\mathsf{Gen}$ *is* errorless-*invertible if* $\mathsf{InvGen}(r, p)$ *outputs either* $\perp$ *(failure symbol) or* $w \in \mathcal{M}$ *for which there exists* $r_G$ *such that* $(r, p) = \mathsf{Gen}(w; r_G)$*.*

In the definition, we consider that $\mathsf{InvGen}$ succeeds in inverting $\mathsf{Gen}$ if it outputs $w'$ from which the input $(r', p)$ can be generated by $\mathsf{Gen}$, and thus $w'$ is not necessarily the same as $w$ from which $p$ was actually generated.

Note that defining $\mathsf{InvGen}$ to be deterministic circuits does not lose the generality. If there exists a randomized circuit $\mathsf{InvGen}$ that inverts $\mathsf{Gen}$ with some probability, then by fixing the coins of $\mathsf{InvGen}$ for which the average performance can be achieved, we can say that there exists a deterministic circuit that inverts $\mathsf{Gen}$ with the same probability.

We show that if a fuzzy extractor has the perfect correctness, we can obtain the errorless invertibility for $\mathsf{Gen}$.

**Lemma 1.** *Let* $(\mathsf{Gen}, \mathsf{Rep})$ *be a fuzzy extractor with error* $0$*. If* $\mathsf{Gen}$ *is* $(s, \eta)$*-invertible, then* $\mathsf{Gen}$ *is* $(s + s_{\text{rep}}, \eta)$*-errorless-invertible, where* $s_{\text{rep}}$ *is the size of circuit* $\mathsf{Rep}$*.*

*Proof.* Let $\mathsf{InvGen}$ be the inverter of $(s, \eta)$-invertibility of $\mathsf{Gen}$. Then, we construct an inverter $\mathsf{InvGen}'$ such that on input $(r, p)$, (1) run $w \leftarrow \mathsf{InvGen}(r, p)$, (2) output $w$ if $\mathsf{Rep}(w, p) = r$, and output $\perp$ otherwise. The correctness property of $(\mathsf{Gen}, \mathsf{Rep})$ guarantees that the output of $\mathsf{InvGen}'$ is a valid inverse of $(r, p)$. □

Since we prove our negative result for computational fuzzy extractors with errorless invertibility, Lemma 1 implies that our negative result can also be applied to computational fuzzy extractors with perfect correctness.

We will prove that the existence of a computational fuzzy extractor implies the existence of an error-correcting code. We provide some notions regarding coding theory.

**Definition 6.** *We say a metric space* $(\mathcal{M}, \mathsf{dis})$ *is* $(s, t)$-bounded-error samplable *if there exists a randomized circuit* $\mathsf{ErrSmp}$ *of size* $s$ *such that for all* $0 \leq t' \leq t$ *and* $w \in \mathcal{M}$*,* $\mathsf{ErrSmp}(w, t')$ *outputs a random point* $w' \in \mathcal{M}$ *satisfying* $\mathsf{dis}(w, w') = t'$*.*

**Definition 7.** *Let* $C$ *be a set over a metric space* $\mathcal{M}$*. We say* $C$ *is a* $(t, \epsilon)$-maximal-error Shannon code *if there exists an efficient recover procedure* $\mathsf{Rec}$ *such that for all* $t' \leq t$ *and* $c \in C$*,* $\Pr[\mathsf{Rec}(\mathsf{ErrSmp}(c, t')) \neq c] \leq \epsilon$*.*

**Definition 8.** *Let* $(\mathcal{M}, \mathsf{dis})$ *be a metric space that is* $(s, t)$*-bounded-error samplable by a circuit* $\mathsf{ErrSmp}$*. For a distribution* $C$ *over* $\mathcal{M}$*, we say* $C$ *is a* $(t, \epsilon)$-average-error Shannon code *if there exists an efficient recover procedure* $\mathsf{Rec}$ *such that for all* $t' \leq t$ *and* $c \in C$*,* $\Pr_{c \in C}[\mathsf{Rec}(\mathsf{ErrSmp}(c, t')) \neq c] \leq \epsilon$*.*

The following fact can be obtained by Markov's inequality. (See [5] for the proof.)

**Lemma 2** ([5])**.** *Let* $C$ *be a* $(t, \epsilon)$*-average-error Shannon code with recovery procedure* $\mathsf{Rec}$ *such that* $\mathrm{H}_\infty(C) \geq k$*. Then, there exists a set* $C'$ *with* $|C'| \geq 2^{k-1}$ *that is* $(t, 2\epsilon)$*-maximal-error Shannon code with recovery procedure* $\mathsf{Rec}$*.*

We prove that if the generation procedure is errorless-invertible, then the existence of a computational fuzzy extractor implies the existence of a maximal-error Shannon code.

**Theorem 1.** *Let $(\mathcal{M}, \mathsf{dis})$ be a metric space that is $(s_{\mathrm{smp}}, t)$-bounded-error samplable. Let $(\mathsf{Gen}, \mathsf{Rep})$ be an $(\mathcal{M}, m, \ell, t, s_{\mathsf{sec}}, \epsilon)$-computational fuzzy extractor with error $\delta$. Let $s_{\mathrm{rep}}$ denote the size of the circuit that computes $\mathsf{Rep}$. If $\mathsf{Gen}$ is $(s_{\mathrm{inv}}, \eta)$-errorless-invertible, and it holds that $s_{\mathrm{sec}} \geq s_{\mathrm{inv}} + ts_{\mathrm{smp}} + (t+1)s_{\mathrm{rep}}$, then there exists a value $p$ and a set $C$ with $|C| \geq 2^{-\log(2^{-\ell}+\frac{\rho}{|\mathcal{M}|})-1}$ that is a $(t, 2\rho)$-maximal-error Shannon code with recovery procedure $\mathsf{InvGen}(\mathsf{Rep}(\cdot, p), p)$, where $\rho = \epsilon + \eta + (t+1)\delta$.*

*Proof.* Let $W$ be an arbitrary distribution on $\mathcal{M}$ of min-entropy $m$. By the security property of the computational fuzzy extractor $(\mathsf{Gen}, \mathsf{Rep})$, we have that $\Delta^{s_{\mathrm{sec}}}((R, P), (U_\ell, P)) \leq \epsilon$ for $(R, P) \leftarrow \mathsf{Gen}(W)$.

Let $\mathsf{InvGen}$ be an inverter of the $(s, 1-\eta)$-errorless-invertibility of $\mathsf{Gen}$. We consider the modified inverter $\mathsf{InvGen}'$:

1. On input $r \in \{0,1\}^\ell$ and $p \in \{0,1\}^*$, compute $w \leftarrow \mathsf{InvGen}(r, p)$.
2. If $w \neq \perp$ and $\mathsf{Rep}(w, p) = r$, output $w$. Otherwise, output a random element in $\mathcal{M}$.

The procedure $\mathsf{InvGen}'$ can be implemented by a circuit of size $s_{\mathrm{inv}} + s_{\mathrm{rep}}$. Define the event $E_{\mathrm{suc}}$ such that

$$E_{\mathrm{suc}} = \{w \neq \perp \wedge \mathsf{Rep}(w, P) = R\},$$

where $(R, P) \leftarrow \mathsf{Gen}(W), w \leftarrow \mathsf{InvGen}(R, P)$. By the correctness property of $(\mathsf{Gen}, \mathsf{Rep})$ and the failure probability of $\mathsf{InvGen}$, we have that $\Pr[E_{\mathrm{suc}}] \geq 1 - (\eta + \delta)$.

Define the following procedure $D$:

1. On input $r \in \{0,1\}^\ell, p \in \{0,1\}^*$, and $t \in \mathbb{N}$, compute $w \leftarrow \mathsf{InvGen}'(r, p)$.
2. For all $1 \leq t' \leq t$, do the following:
   (a) $w' \leftarrow \mathsf{ErrSmp}(w, t')$.
   (b) If $\mathsf{Rep}(w', p) \neq r$, output 0. Otherwise, do nothing.
3. Output 1.

The procedure $D$ "efficiently" checks whether $\mathsf{Rep}$ can correctly output the string $r$ from the corresponding $p$ and $w$ with random $t$-bounded errors. We need the efficiency of $D$ since otherwise the "error-correcting" property of $\mathsf{Rep}$ may not be taken over from the computational security of $(\mathsf{Gen}, \mathsf{Rep})$.

The procedure $D$ can be implemented by a circuit of size $s_{\mathrm{inv}}+ts_{\mathrm{smp}}+(t+1)s_{\mathrm{rep}}$. Note that in the procedure $D$, we can easily check whether the event $E_{\mathrm{suc}}$ occurs or not (by checking that a random element is produced in $\mathsf{InvGen}'$). Thus, by the invertibility of $\mathsf{Gen}$ and the correctness property of $(\mathsf{Gen}, \mathsf{Rep})$, we have that $\Pr[D(R, P, t) = 1 \wedge E_{\mathrm{suc}}] \geq 1-(\eta+(t+1)\delta)$. Since $\Delta^{s_{\mathrm{sec}}}((R, P), (U_\ell, P)) \leq \epsilon$, if $s_{\mathrm{sec}} \geq s_{\mathrm{inv}} + ts_{\mathrm{smp}} + (t+1)s_{\mathrm{rep}}$, it holds that

$$\begin{aligned}\Pr[D(U_\ell, P, t) = 1 \wedge E_{\mathrm{suc}}] &\geq 1 - (\epsilon + \eta + (t+1)\delta) \\ &= 1 - \rho.\end{aligned}$$

By the averaging argument, there exists a value $p$ such that $\Pr[D(U_\ell, p, t) = 1 \wedge E_{\mathrm{suc}}] \geq 1 - \rho$. This implies that, for all $1 \leq t' \leq t$,

$$\Pr\left[\begin{array}{l} w \leftarrow \mathsf{InvGen}'(R, p), \\ w' \leftarrow \mathsf{ErrSmp}(w, t') \end{array} : \mathsf{Rep}(w', p) = R \wedge E_{\mathrm{suc}}\right] \geq 1 - \rho, \tag{1}$$

where $R = U_\ell$. Since the event $E_{\mathrm{suc}}$ implies that $\mathsf{InvGen}(R, p) = w$, we have that, for all $1 \leq t' \leq t$,

$$\Pr\left[\begin{array}{l} w \leftarrow \mathsf{InvGen}'(U_\ell, p), \\ w' \leftarrow \mathsf{ErrSmp}(w, t') \end{array} : \mathsf{InvGen}(\mathsf{Rep}(w', p), p) = w\right] \geq 1 - \rho.$$

This implies that a distribution $\mathsf{InvGen}'(U_\ell, p)$ is a $(t, \rho)$-average-error Shannon code with recovery procedure $\mathsf{InvGen}(\mathsf{Rep}(\cdot, p), p)$. By applying Lemma 2, we can show that there is a set $C$ with $|C| \geq 2^{k-1}$ that is a $(t, 2\rho)$-maximal-error Shannon code for $k \leq \mathrm{H}_\infty(\mathsf{InvGen}'(U_\ell, p))$.

Finally, we prove that $\mathrm{H}_\infty(\mathsf{InvGen}'(U_\ell, p)) \geq -\log(2^{-\ell} + \frac{\rho}{|\mathcal{M}|})$. Define

$$R_{\mathrm{good}} = \left\{ r \in \{0,1\}^\ell : \begin{array}{l} w \leftarrow \mathsf{InvGen}(r, p), \\ w \neq \perp \wedge \mathsf{Rep}(w, p) = r \end{array} \right\}.$$

By equation (1), it holds that $|R_{\mathrm{good}}| \geq (1 - \rho)2^\ell$. Let $W_{\mathrm{good}} = \{\mathsf{InvGen}(r, p) : r \in R_{\mathrm{good}}\}$. By the definition of $\mathsf{InvGen}'$, we have that

$$\Pr[\mathsf{InvGen}'(U_\ell, p) = w] = \begin{cases} 2^{-\ell} + \frac{\rho}{|\mathcal{M}|} & w \in \mathcal{M} \cap W_{\mathrm{good}} \\ \frac{\rho}{|\mathcal{M}|} & w \in \mathcal{M} \setminus W_{\mathrm{good}}. \end{cases}$$

Therefore, the min-entropy of $\mathsf{InvGen}'(U_\ell, p)$ is $-\log(2^{-\ell} + \frac{\rho}{|\mathcal{M}|})$. □

It is known that a secure sketch can be constructed from a Shannon code, which is explicitly presented in [5], and implicitly stated in [3, Section 8.2].

**Lemma 3** ([3, 5])**.** *For an alphabet $Z$, let $C$ be a $(t, \delta)$-maximal-error Shannon code over $Z^n$. Then, there exists a $(Z^n, m, m - (n \log |Z| - \log |C|), t)$ secure sketch with error $\delta$ for the Hamming metric over $Z^n$.*

An information-theoretic fuzzy extractor can be constructed from a secure sketch and a strong extractor [3]. In particular, if we use universal hashing as strong extractor, we obtain the following result.

**Lemma 4** ([3])**.** *Let $(\mathsf{SS}, \mathsf{Rec})$ be an $(\mathcal{M}, m, \tilde{m}, t)$-secure sketch with error $\delta$, and $\mathsf{Ext}$ an $(n, \tilde{m}, \ell, \epsilon)$-strong extractor given by universal hashing (any $\ell \leq \tilde{m} - 2\log(\frac{1}{\epsilon}) + 2$ can be achieved). Then, the following $(\mathsf{Gen}, \mathsf{Rep})$ is an $(\mathcal{M}, m, \ell, t, \epsilon)$-fuzzy extractor:*

- $\mathsf{Gen}(w; r, x)$ *: set $P = (\mathsf{SS}(w; r), x)$, $R = \mathsf{Ext}(w; x)$, and output $(R, P)$.*
- $\mathsf{Rep}(w', (s, x))$ *: recover $w = \mathsf{Rec}(w', s)$ and output $R = \mathsf{Ext}(w; x)$.*

By combining Theorem 1 and Lemmas 3 and 4, we obtain the following corollary.

**Corollary 1.** *Let $Z$ be an alphabet. Let* $(\mathsf{Gen}, \mathsf{Rep})$ *be a* $(Z^n, m, \ell, t, s_{\mathsf{sec}}, \epsilon)$*-computational fuzzy extractor with error* $\delta$*. Let* $s_{\mathrm{rep}}$ *denote the size of the circuit that computes* $\mathsf{Rep}$*. If* $\mathsf{Gen}$ *is* $(s_{\mathrm{inv}}, \eta)$*-errorless-invertible, and it holds that* $s_{\mathrm{sec}} \geq s_{\mathrm{inv}} + tn\log|Z| + (t+1)s_{\mathrm{rep}}$*, then there exists a* $(Z^n, m, \ell, t, \epsilon')$ *(information-theoretic) fuzzy extractor with error* $2\rho$ *for any* $\ell \leq m - n\log|Z| - \log(2^{-\ell} + \frac{\rho}{|Z|^n}) - 2\log(\frac{1}{\epsilon'}) + 1$*, where* $\rho = \epsilon + \eta + (t+1)\delta$*.*

In particular, in the above corollary, if we choose $m = n\log|Z|$ and $\frac{\rho}{|Z|^n} \leq 2^{-\ell}$, then a $(Z^n, n\log|Z|, \ell, t, s_{\mathsf{sec}}, \epsilon)$-computational fuzzy extractor implies a $(Z^n, n\log|Z|, \ell - 2\log(\frac{1}{\epsilon'}), t, \epsilon')$-fuzzy extractor with error $2\rho$.

As in the negative result of [5], we do not claim about the efficiency of the resulting fuzzy extractor. In our case, the non-explicit parts are (1) fixing the value $p$, and (2) constructing a maximal-error Shannon code from an average-error one (Lemma 2) in Theorem 1.

## Acknowledgments

The authors are grateful to Masahiro Mambo for his helpful comments.

This work was supported in part by JSPS Grant-in-Aid for Scientific Research Numbers 23500010, 23700010, 24240001, and 25106509.

## References

[1] X. Boyen. Robust and reusable fuzzy extractor. In P. Tuyls, B. Skoric, and T. Kevenaar, editors, *Security with Noisy Data*, pages 101–112. Springer, 2007.

[2] M. Cheraghchi, F. Didier, and A. Shokrollahi. Invertible extractors and wiretap protocols. *IEEE Transactions on Information Theory*, 58(2):1254–1274, 2012.

[3] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008.

[4] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors. In P. Tuyls, B. Skoric, and T. Kevenaar, editors, *Security with Noisy Data*, pages 79–99. Springer, 2007. An updated version is available at http://www.cs.bu.edu/~reyzin/fuzzysurvey.html.

[5] B. Fuller, X. Meng, and L. Reyzin. Computational fuzzy extractors. In K. Sako and P. Sarkar, editors, *ASIACRYPT (1)*, volume 8269 of *Lecture Notes in Computer Science*, pages 174–193. Springer, 2013.

[6] J. Radhakrishnan and A. Ta-Shma. Bounds for dispersers, extractors, and depth-two superconcentrators. *SIAM J. Discrete Math.*, 13(1):2–24, 2000.