Privacy-preserving datasets of eye-tracking samples with applications in XR: Supplementary Material

Brendan David-John, *Member, IEEE*, Kevin Butler *Senior Member, IEEE* and Eakta Jain, *Member, IEEE*

1 Threat Scenario *k*-anonymity Details

Age and gender demographics are generalized by grouping values into ranges to achieve *k*-anonymity. The number of data rows for each unique combination of age and gender ranges must be *k* or greater to maintain the privacy guarantee. The combined dataset of ET-DK2 and 360_em consists of 24 individuals with age and gender values listed in Table 1.

Table 1: Age and Gender demographics for ET-DK2 and 360_em datasets. Note that Subject ID 1 from both datasets were excluded from analysis due to data loss and subject sickness during data collection, respectively.

Dataset	Subject ID	Age	Gender
ET-DK2	2	M	43
ET-DK2	3	F	27
ET-DK2	4	M	29
ET-DK2	5	M	32
ET-DK2	6	F	28
ET-DK2	8	M	26
ET-DK2	9	F	23
ET-DK2	10	M	30
ET-DK2	11	F	28
ET-DK2	12	M	26
ET-DK2	13	M	52
ET-DK2	14	M	26
ET-DK2	15	M	35
ET-DK2	16	M	50
ET-DK2	17	M	33
ET-DK2	18	M	31
ET-DK2	19	M	32
ET-DK2	20	M	36
360_em	2	M	38
360_em	3	M	29
360_em	4	F	23
360_em	5	F	31
360_em	6	M	27
360_em	7	M	31
360_em	8	F	23
360_em	9	M	24
360_em	10	M	23
360_em	11	M	27
360_em	12	M	23
360_em	13	M	23
360_em	14	M	32

Ranges were selected for each value of *k* that maximized the total number of groups while ensuring each group had at least *k* rows matching the ranges of age and gender. The ranges of age and gender used to establish *k*-anonymity are listed in Table 2.

Table 2: Gender and age ranges used to generalize the ET-DK2 and 360_em demographics for *k*-anonymity. For each value of *k* the data rows are mapped into the listed ranges based on actual values. For example, (Male, 23-31) would be assigned to all Males between the age of 23 and 31. Male/Female refers to the data rows not specifying either value for Gender.

k	Gender & Age Generalization
4	(Female, 23-31), (Male, 23-27), (Male, 29-31), (Male, 32-33), (Male, 35-52)
6	(Female, 23-31), (Male, 23-27), (Male, 29-33), (Male, 35-52)
8	(Male/Female, 23-27), (Male/Female, 28-31), (Male/Female, 32-52)
15	(Male/Female, 23-28), (Male/Female, 29-52)

2 Privacy Mechanism Pseudocode

2.1 k-same-synth

```
procedure k-SAME-SYNTH(k, sample_data, fix_event_params, sacc_event_params)
Parameters: k - k-anonymity parameter
            sample_data - Time series of gaze samples, indexed by stimulus m, identity i, and fixation/saccade events e
            fix_event_params - Fixation Gaussian parameters, indexed by stimulus m, identity i, and event e
            sacc_event_params - Velocity profile parameters, indexed by stimulus m, identity i, and event e
    fix_event_params ← k-same-select sequence(k, fix_event_params)                 ▷ Make fixation params k-anonymous
    sacc_event_params ← k-same-select-sequence(k, sacc_event_params)               ▷ Make saccade params k-anonymous
    for m = 1 to num_stimuli do                                                   ▷ Process events from each stimulus independently
        for i = 1 to num_identities do                                            ▷ Process samples for each identity
            fix_data_params ← fix_event_params[m, i, :]                           ▷ List of fixation event parameters
            for e = 1 to num_fixations do
                μx, μy, σx, σy, t ← fix_data_params[e]
                sample_data[m, i, e] ← SynthFixation(μx, μy, σx, σy, t)   ▷ Synthesize samples for fixation e by sampling 2D Normal distribution
            sacc_data_params ← sacc_event_params[m, i, :]                         ▷ List of saccade event parameters
            for e = 1 to num_saccades do
                a, b, c, t ← sacc_data_params[e]
                sample_data[m, i, e] ← SynthSaccade(a, b, c, t)   ▷ Synthesize samples for saccade e using velocity profile from Gaussian model
    return sample_data
```

2.2 event-synth-PD

```
procedure EVENT-SYNTH-PD(k, γ, sample_data, fix_event_params, sacc_vel_profiles, CVAE_enc, CVAE_dec)
Parameters: k, γ - plausible deniability parameters
            sample_data - Time series of gaze sample, indexed by stimulus m, identity i, and fixation/saccade events e
            fix_event_params - Fixation Gaussian parameters, indexed by stimulus m, identity i, and event e
            sacc_vel_profiles - Saccade velocities and conditions, indexed by stimulus m, identity i, and event e
            CVAE_enc - Encoder network of C-VAE, maps input to latent space distributions defined by μ and σ
            CVAE_dec - Decoder network of C-VAE, maps input random samples z⊕c to synthetic velocities
    for m = 1 to num_stimuli do                                                   ▷ Process events from each stimulus independently
        for i = 1 to num_identities do                                            ▷ Process samples for each identity
            fix_data_params ← fix_event_params[m, i, :]                           ▷ List of fixation event parameters
            for e = 1 to num_fixations do                                         ▷ Synthesize fixation samples until PD criterion is met
                d = (μx, μy, σx, σy, t) ← fix_data_params[e]                       ▷ Params for fixation e
                M_fix ← N(x, y)                                                   ▷ 2D Normal distribution that returns t values
                result ← False
                while result == False do
                    y ← M_fix(d)                                                  ▷ Generate t samples from distribution with curr params
                    Pr_d ← Pr{y ← M_fix(d)}                                       ▷ Probability real seed generated synthetic samples y
                    result ← PD Event Privacy Test(k, γ, Pr_d, M_fix, fix_event_params[m, ≠i, :])   ▷ ≠i indicates all individual data besides i
                sample_data[m, i, e] ← y
            sacc_data ← sacc_vel_profiles[m, i, :]                                ▷ List of real data saccade profiles
            for e = 1 to num_saccades do                                          ▷ Synthesize fixation samples until PD criterion is met
                d = (μ1, σ1, ···, μL, σL) ← C−VAE_enc(sacc_data[e])
                M_sacc ← N1, ···, NL                                              ▷ Define M as L independent Normal distributions
                result ← False
                while result == False do
                    y = (z1, ···, zL) ← M_sacc(d)
                    Pr_d ← Pr{y ← M_sacc(d)}                                      ▷ Probability real seed generated synthetic samples y
                    result ← PD Event Privacy Test(k, γ, Pr_d, M_sacc, sacc_vel_profiles[m, ≠i, e])   ▷ ≠i indicates all individual data besides i
                sample_data[m, i, e] ← y
    return sample_data
```

```
procedure PD EVENT PRIVACY TEST(k, γ, Pr_d, M, D))
Parameters: k, γ - plausible deniability parameters, Pr_d - Probability of real seed for y, Pr{y ← M(d)}
            M - generative model that synthesized y, D - data records from identities other than input
    i′ ← unique integer i′, s.t. γ^(−i′−1) < Pr_d ≤ γ^(−i′)
    k′ ← 0
    for i = 1 to num_identities do
        D_i ← D[i]
        for d_a ∈ D_i do
            if γ^(−i′−1) < Pr{y = M(d_a)} ≤ γ^(−i′) then
                k′ ← k′ + 1
                Break                                                             ▷ Move for loop for i onto the next identity
    if k′ ≥ k − 1 then return Pass
    else return Fail
```

2.3 Kalεido

The pseudocode below details the kalεido approach for a stream of n_{raw} gaze samples $g_{1,\cdots,n_{raw}}$, window size w, privacy parameter ε, sample distance threshold l_{thresh}, sample skipping parameter t_{skip}, spatial parameter r, and ratio of testing to publishing privacy budget h.

The adaptive algorithm includes several parameters that allow for privacy budget savings while processing the gaze sample at each timestamp. First, a fixed time duration $t_{skip} = 50ms$ is used to skip gaze samples that arrive within t_{skip} of the last published gaze position. Next, after t_{skip} has passed since the last published gaze point, the algorithm moves on to the testing phase. If the current gaze position is within the fixation threshold determined by l_{thresh} and ε^{test}, then the previously published position is re-used, and only ε^{test} of the budget for the current time window is consumed. The algorithm enters the publishing phase if the new gaze position is farther than the threshold. A noisy gaze position is generated using the ε^{pub} budget with a Planar Laplacian mechanism [1]. The amount of the ε^{pub} budget used decreases adaptively to preserve as much utility as possible while maintaining ε-DP guarantee within each time window. This process is repeated for each time window, and any leftover ε^{pub} budget is recycled into the next window. A complete description of the proof that each window consumes at most ε of the privacy budget is available in the original paper [2].

procedure KALεIDO DP($g_{1,\cdots,n_{raw}}$, w, ε, l_{thresh}, t_{skip}, r, h)
Parameters: $g_{1,\cdots,n_{raw}}$ - Stream of gaze positions, w - Window size (# samples), ε - DP privacy level
l_{thresh} - Distance threshold for testing, t_{skip} - # of samples to skip over during testing
r - Privacy radius for DP, h - Ratio of privacy budget used for testing
  $n_{test} \leftarrow \lceil w/t_{skip} \rceil$ ▷ Number of points to test for each window
  $\varepsilon_{test} \leftarrow \varepsilon/(h \cdot n_{test})$ ▷ Privacy budget allocated to test each sample
  $i_{test} \leftarrow null$ ▷ Index of the last tested gaze position.
  $i_{pub} \leftarrow null$ ▷ Index of the last published gaze position.
  $g'_i \leftarrow zeros(n_{raw})$ ▷ Published gaze position for sample i, initialized to zeros.
  $\varepsilon_i^{pub} \leftarrow zeros(n_{raw})$ ▷ List of privacy budget consumed for sample i, initialized to zeros.
  for $i = 1$ to num_raw **do** ▷ Process each window of raw gaze samples
    if $i_{test} \neq null$ AND $t(i) - t(i_{test}) < t_{skip}$ **then** ▷ Check if sample should be skipped based on t_{skip} parameter
      $g'_i \leftarrow g'_{i_{pub}}$
      $\varepsilon_i^{pub} \leftarrow 0$
      Continue
    $i_{test} = i$
    $l_{dis} = d(g_i, g'_{i_{pub}})$ ▷ Distance between gaze sample i and last published
    $\eta \sim Lap(1/\varepsilon_{test})$ ▷ Sample from Laplace distribution, small values of ε_{test} introduce more noise
    if $l_{dis} \neq null$ AND $l_{dis} \leq l_{thresh} + \eta$ **then** ▷ Test if current gaze is close enough to last published to repeat
      $g'_i \leftarrow g'_{i_{pub}}$
      $\varepsilon_i^{pub} \leftarrow 0$
      Continue
    $i_{pub} \leftarrow i$ ▷ Publish a new gaze sample, update index of last published
    $\varepsilon_{rem} \leftarrow \varepsilon - \varepsilon/h - \sum_{k=i-n_{raw}+1}^{i-1} \varepsilon_k^{pub}$ ▷ Compute remaining privacy budget for this window
    $\varepsilon_i^{pub} \leftarrow \varepsilon_{rem}/2$
    $g'_i \leftarrow PlanarLap(g_i, \varepsilon_i^{pub}/r)$
  return g'

3 C-VAE Model Training Procedure

The C-VAE model for generating synthetic saccade profiles was trained using tensorflow version 1.13.1. Models were trained independently for each dataset using data from all individuals and stimuli. Training was performed using 75% of the available data with the remaining 25% used as a validation set.

All models were trained with an ADAM optimizer using tensorflow's Model compile and fit functions. The loss function was defined as

$$\mathbf{L}(x, \mathbf{D}(z)) = ||x - \mathbf{D}(z)||_2 - \mathbf{KL}(\mathbf{N}(\mu, \sigma), \mathbf{N}(0,1)),$$

where the first term is Mean Squared Error for the reconstructed synthetic profile and the second terms employs KL Divergence to enforce latent space sampling that follows a normal distribution with zero mean.

4 C-VAE Model Hyper-Parameter Optimization

Hyper-parameters were tuned using the EHTask dataset as it contained a longer duration of data compared to the DGaze dataset. Grid search optimization was performed over the following sets of values, with optimal parameters in bold:

- Learning Rate: 0.001, **0.01**
- Batch Size: 20, 60, **100**
- Number of Epochs: 10, **20**, 30
- Encoder Hidden Layer with ReLU activation function: **32**, 64, 96 Nodes
- Latent Space Dimension: 32, **64**, 96
- Decoder Hidden Layer with linear activation function: 32, 64, **96** Nodes

The optimal parameters produced an average loss of 0.33 on the validation set.

Acknowledgments

Authors acknowledge funding from the National Science Foundation (Awards FWHTF-2026540, CNS-1815883, and CNS-1562485), the National Science Foundation GRFP (Awards DGE-1315138 and DGE-1842473), and the Air Force Office of Scientific Research (Award FA9550-19-1-0169).

References

[1] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 901–914, 2013.

[2] J. Li, A. R. Chowdhury, K. Fawaz, and Y. Kim. Kalεido: Real-time privacy control for eye-tracking systems. In *29th USENIX Security Symposium (USENIX Security 20)*, 2020.