Surface Urban Heat Island Intensity (SUHII)

Analysis Tool

Technical Documentation

Nishan Sah
Sakina Mammadova
Nandita Kannapadi
Gabe Hafemann
Zoë Baker

December 2025

Contents

1 Introduction

The Surface Urban Heat Island Intensity (SUHII) Analysis Tool is a Python-based framework for quantifying the urban heat island effect using satellite-derived Land Surface Temperature (LST) data. The tool implements multiple peer-reviewed methodologies for defining urban and rural reference areas, enabling researchers to select approaches appropriate for their study context and compare results across methods.

1.1 Purpose and Scientific Basis

Urban Heat Islands (UHIs) represent one of the most significant anthropogenic modifications to local climate. The energetic basis of this phenomenon is characterized by increased heat storage in the urban fabric and reduced latent heat flux due to vegetation loss, which has been well established in the literature.

This tool calculates Surface UHI Intensity (SUHII) using satellite thermal imagery (LST), which is distinct from Canopy UHI (air temperature). As noted by Zhou et al. (2014), SUHII exhibits distinct spatial patterns driven by surface cover, albedo, and anthropogenic heat, necessitating rigorous spatial definition of "urban" and "rural" baselines.

Quantifying UHI intensity is essential for:

- Urban climate adaptation planning
- Sustainable Development Goal 11 (Sustainable Cities) monitoring
- Heat-health early warning systems
- Urban greening intervention assessment

1.2 Design Philosophy

The tool was developed with three guiding principles:

1. **Methodological Transparency**: All algorithms are traceable to peer-reviewed literature with explicit citations.
2. **Flexibility**: Multiple methods are implemented to accommodate different data availability scenarios and research questions.
3. **Reproducibility**: Configuration-driven execution ensures analyses can be exactly replicated.

1.3 Scope

This tool calculates **Surface** UHI Intensity (SUHII) using satellite thermal imagery, distinct from **Canopy** UHI Intensity (CUHII) measured by air temperature sensors. SUHII represents the radiative temperature difference between urban and rural land surfaces as observed from space.

2 Theoretical Background

2.1 Defining Urban Heat Island Intensity

The Surface Urban Heat Island Intensity is defined as the difference between mean urban and mean rural land surface temperatures:

$$SUHII = \overline{LST}_{urban} - \overline{LST}_{rural} \quad (1)$$

Where:

- $\overline{LST}_{urban}$ = Mean LST of pixels classified as urban
- $\overline{LST}_{rural}$ = Mean LST of pixels in the rural reference area

This absolute difference formulation is the standard approach used in recent literature, including Raj & Yun (2024) and Fernandes et al. (2024). An alternative normalized formulation is used by Ahmad et al. (2024)—see Section 3.4.

2.2 Challenges in SUHII Estimation

Three fundamental challenges affect SUHII estimation:

2.2.1 Urban Definition Ambiguity

"Urban" can be defined by administrative boundaries, population density, land cover classification, or spectral indices. Each definition yields different urban extents and consequently different SUHII values.

2.2.2 Rural Reference Selection

The magnitude of SUHII is highly sensitive to the definition of the rural reference. Raj & Yun (2024) demonstrated that for highly urbanized cities (urban fraction > 50%), the choice between a surrounding buffer method and an in-city non-urban method can alter SUHII estimates by over 1.0°C. Their study compared buffer-based and in-city methods across five South Korean metropolitan areas.

2.2.3 Confounding Factors

Elevation Bias: Mentaschi et al. (2022) and Raj & Yun (2024) highlight that elevation differences induce temperature biases via the environmental lapse rate, necessitating correction or filtering. Raj & Yun (2024) address this by filtering rural pixels to within ±50m of urban average elevation.

Temporal Mismatch: LULC products may not match LST acquisition date, introducing classification errors in rapidly changing urban environments.

3 Methodology

3.1 Urban Area Definition

The tool implements three methods for defining urban pixels, selectable via the `urban_selection.method` configuration parameter.

3.1.1 LULC-Based Method (`method: "lulc"`)

Urban pixels are identified using land use/land cover classification products. The user specifies which class codes represent urban/built-up areas.

Algorithm:

```
urban_mask = pixel in {urban_classes}
```

Applicable LULC Products:

Product	Urban Class Code	Resolution
Dynamic World	6 (Built Area)	10m
ESA WorldCover	50 (Built-up)	10m
MODIS MCD12Q1	13 (Urban)	500m
Copernicus GLC	50 (Built-up)	100m

Table 1: Applicable LULC products for urban classification

Scientific Rationale for Dynamic World:

While static products (e.g., ESA WorldCover) are supported, the tool explicitly recommends and supports the Dynamic World dataset as a superior alternative for dynamic urban analysis. Dynamic World is a near-real-time (NRT) 10m resolution global land use land cover dataset generated using a deep learning model (Fully Convolutional Neural Network) applied to Sentinel-2 imagery (Brown et al., 2022).

The justification for using this advanced product over traditional static maps is threefold:

1. **Methodological Robustness:** Recent literature validates the superiority of machine learning and deep learning approaches for urban LULC classification. Vignesh et al. (2022) demonstrated that Deep Learning models (specifically Long Short-Term Memory Recurrent Neural Networks) applied to multispectral satellite data (Landsat-8) significantly outperform traditional classifiers in capturing complex urban heterogeneity. Similarly, Kumar et al. (2025) validated the efficacy of machine learning (Random Forest) on Sentinel-2 imagery within Google Earth Engine for accurate urban mapping.
2. **Temporal Synchronization:** SUHII analysis requires precise temporal alignment between the thermal image (LST) and the land cover mask. Traditional annual composites may fail to capture rapid urbanization or seasonal phenology changes.
3. **Probabilistic Definition:** Unlike binary classification products, Dynamic World provides probability scores for each class. This allows the tool to define "Built Area" based on a confidence threshold (e.g., built_probability > 0.5), offering greater sensitivity in mixed-pixel transition zones common in peri-urban areas.

Literature Examples:

- Raj & Yun (2024) use Landsat-8 land cover classification to identify urban pixels in their South Korean study
- Mentaschi et al. (2022) use GHSL (Global Human Settlements Layer) with >15% built-up probability threshold in their global analysis

3.1.2 NDVI-Based Method (`method: "ndvi"`)

Urban pixels are identified using a Normalized Difference Vegetation Index (NDVI) threshold, based on the assumption that built-up areas have low vegetation cover.

Algorithm:

```
urban_mask = NDVI < ndvi_max_threshold
```

Default Threshold: 0.3

Scientific Basis:

While most studies use LULC products, Ahmad et al. (2024) and Keerthi Naidu & Chundeli (2023) demonstrate a strong, statistically significant inverse correlation between LST and NDVI

(Pearson $r \approx -0.6$ to -0.8 in summer). Ahmad et al. (2024) report correlations of $r = -0.673$ in summer for Delhi. This strong linearity supports the use of low NDVI as a proxy for built-up/impervious surfaces when high-resolution LULC data is unavailable.

Furthermore, Tempa et al. (2024) confirm the robustness of NDVI as a monitoring tool for urban transitions, showing that significant drops in NDVI (e.g., 75% loss in "very healthy vegetation") directly correspond to built-up expansion in rapidly urbanizing regions.

Warning: This method should only be used when LULC data is unavailable. Results should be interpreted with caution in semi-arid regions where bare soil is prevalent.

3.1.3 Combined LULC + NDVI Method (`method: "lulc_ndvi"`)

This hybrid approach uses LULC classification to identify built-up areas, then applies an additional NDVI filter to exclude any remaining vegetated pixels within those areas (e.g., urban parks, gardens, tree-lined streets).

Algorithm:

```
urban_mask = (pixel in {urban_classes}) AND (NDVI < ndvi_max_threshold)
```

Rationale: Urban parks and green spaces, while administratively urban, exhibit thermal behavior more similar to rural vegetation than impervious surfaces. Including them in the urban sample would underestimate true SUHII for built infrastructure.

Scientific Basis:

Zhou et al. (2014) emphasize that the "urban" thermal signal is driven by impervious surfaces. Mentaschi et al. (2022) exclude water from their urban definition to avoid thermal contamination. Yang & Yao (2022) demonstrate that even within urban boundaries, vegetated patches maintain distinct lower LSTs. This method synthesizes these findings by strictly isolating the impervious fraction of the LULC urban class, ensuring the urban temperature mean is not biased by urban green spaces, which Keerthi Naidu & Chundeli (2023) show can act as cool islands.

Recommendation: This is the method can be used when both LULC and NDVI data are available, and LULC alone is not enough to filter the vegetated pixels.

3.1.4 Water Body Exclusion

All methods optionally exclude water bodies from the urban mask when `filters.mask_water: true`. Water has distinct thermal properties (high heat capacity, evaporative cooling) that would bias urban LST estimates.

Raj & Yun (2024) and Mentaschi et al. (2022) explicitly exclude water bodies from both urban and rural reference areas in their methodologies.

3.2 Rural Reference Selection

The rural reference area defines the "baseline" temperature against which urban temperatures are compared. Four methods are implemented, selectable via `rural_selection.method`.

3.2.1 Fixed Buffer Method (`method: "buffer"`)

Primary Reference: Raj & Yun (2024)

A fixed-width annular buffer is created around the urban boundary. Rural pixels are selected from within this buffer based on vegetation and elevation criteria.

Algorithm:

```
rural_geometry = Buffer(urban_boundary, width) - urban_boundary
```

Parameters:

- `fixed_width_m`: Buffer width in meters (default: 10,000m)

Raj & Yun (2024) use a buffer with "ichimyeonjuk roughly equivalent to the city area" (∼10 km) for South Korean metropolitan cities, combined with elevation and land cover filtering. Zhou et al. (2014) also utilized an equal-area buffer approach for 32 major Chinese cities, validating the geometric consistency of annular buffers for comparative analysis.

3.2.2 Urban Halo Method (`method: "halo"`)

This method recognizes that the immediate periphery of cities often experiences thermal contamination from the urban heat island "footprint." If the study area is significantly surrounded by built-up spaces, an inner exclusion zone can be skipped before the rural buffer begins.

Algorithm:

```
inner_ring = Buffer(urban_boundary, min_distance)
outer_ring = Buffer(urban_boundary, min_distance + width)
rural_geometry = outer_ring - inner_ring
```

Parameters:

- `min_distance_from_edge_m`: Inner exclusion distance (default: 0m)
- `fixed_width_m`: Buffer width after exclusion zone

Scientific Basis:

Mentaschi et al. (2022) employ a large (70km) kernel to ensure the reference is taken far beyond the "urban footprint," acknowledging that the thermal impact of a city extends beyond its physical boundary. This tool implements that concept as a user-configurable exclusion zone (`min_distance_from_edge_m`), allowing users to replicate the "far-field" reference approach necessary to avoid thermal contamination from suburban sprawl or advection.

3.2.3 Three-Ring Method (`method: "three_rings"`)

Primary Reference: Fernandes et al. (2024)

This method creates dynamically-sized buffer zones scaled to city area, enabling consistent comparisons across cities of different sizes. Only one of the buffers can be applied at a time. Three concentric zones are defined:

1. **Ua (Urban Adjacent):** Immediate urban periphery
2. **FUa (Future Urban Adjacent):** Transition zone
3. **PUa (Peri-Urban):** Rural reference zone

Formulas from Fernandes et al. (2024):

$$W_{Ua} = 0.25\sqrt{A} \tag{2}$$

$$W_{FUa} = 0.25\sqrt{A + A_{Ua}} \tag{3}$$

$$W_{PUa} = 1.5\sqrt{A} - W_{FUa} - W_{Ua} \tag{4}$$

Where:

- A = Urban area (km^2)
- A_{Ua} = Area of the Ua ring (km^2)
- W = Width of each zone (km)

3.2.4 In-City Method (`method: "incity"`)

Primary References: Raj & Yun (2024); Ahmad et al. (2024)

Rural reference pixels are selected from within the administrative boundary itself, representing non-urban land cover (parks, urban forests, undeveloped land) inside the city.

Raj & Yun (2024) refer to this as “Method 1: Non-urban areas within city limits” and compare it directly against the buffer approach (“Method 2”). They use Landsat-8 classification to identify non-urban pixels within city boundaries and find it yields lower SUHII values for highly urbanized cities like Seoul.

Algorithm:

```
rural_geometry = urban_boundary  # Same as urban geometry
rural_mask = rural_geometry AND (NOT urban_pixels) AND vegetation_criteria
```

Normalized UHI Calculation:

When using the incity method, the tool additionally calculates the Normalized UHI following Ahmad et al. (2024):

$$UHI_{normalized} = \frac{\overline{LST}_{urban} - \overline{LST}_{study}}{\sigma_{study}} \tag{5}$$

Where:

- $\overline{LST}_{urban}$ = Mean LST of urban-classified pixels
- $\overline{LST}_{study}$ = Mean LST of entire study area (administrative boundary)
- σ_{study} = Standard deviation of LST across study area

Ahmad et al. (2024) use this normalized formulation in their Delhi study, expressing UHI as standard deviations from the mean rather than absolute temperature difference.

Key Finding from Raj & Yun (2024): Cities with >50% urban land cover show >1°C difference between buffer and incity methods. For less urbanized cities (<40% urban), both methods produce comparable results.

3.2.5 Rural Pixel Filtering

Regardless of geometry method, rural pixels undergo additional filtering:

1. **Vegetation Filter:** NDVI ≥ `vegetation_ndvi_threshold` (default: 0.2)
2. **Water Exclusion:** Remove water bodies
3. **LULC Exclusion:** Remove urban LULC classes from buffer (optional)
4. **Nodata Exclusion:** Remove pixels with invalid LULC values
5. **Distance Filter:** Minimum distance from urban edge (optional)
6. **Elevation Filter:** Within tolerance of urban mean elevation (see Section 3.3)

3.2.6 Water Body Exclusion from Rural Reference

Water bodies are excluded from the rural reference area when `mask_water: true`. This filter is critical for accurate SUHII estimation.

Scientific Rationale:

Water bodies exhibit fundamentally different thermal behavior compared to vegetated land surfaces due to three key physical properties:

1. **High Thermal Inertia:** Water has a specific heat capacity approximately 4× greater than soil and vegetation. This causes water surfaces to warm and cool much more slowly than surrounding land.
2. **Evaporative Cooling:** Open water surfaces experience continuous latent heat flux through evaporation, which suppresses surface temperatures relative to vegetated land under equivalent radiative forcing.
3. **Low Surface Emissivity Variation:** Unlike vegetated surfaces where emissivity varies with moisture content and vegetation type, water maintains relatively constant emissivity (∼0.98).

Raj & Yun (2024) and Mentaschi et al. (2022) explicitly exclude water bodies from both urban and rural reference areas in their methodologies.

3.3 Elevation Correction

Primary References: Raj & Yun (2024); Mentaschi et al. (2022)

Elevation differences between urban and rural areas introduce systematic temperature biases due to the environmental lapse rate (∼6.5°C per 1000m elevation gain). The tool implements adaptive elevation filtering to ensure climatic comparability.

Raj & Yun (2024) enforce a strict filter, excluding rural pixels outside ±50m of the urban average elevation. Mentaschi et al. (2022) define the standard environmental lapse rate of −6.5 K/km. Yang & Yao (2022) further validate this necessity in their global analysis of 346 cities, demonstrating that failing to control for elevation in rural reference selection introduces significant bias.

3.3.1 Reference Elevation Calculation

The reference elevation is calculated as the mean elevation of urban-classified pixels:

$$\overline{E}_{urban} = \frac{1}{n}\sum_{i=1}^{n} DEM_i \quad \text{where } i \in \text{urban pixels} \tag{6}$$

3.3.2 Adaptive Tolerance Algorithm

This tool implements the Raj & Yun (2024) filtering approach but makes it adaptive. Rural pixels are filtered to those within an elevation tolerance of the urban reference. The algorithm iteratively expands tolerance until a minimum pixel count is reached:

```
current_tolerance = initial_tolerance
while current_tolerance <= max_tolerance:
    candidate_pixels = rural_pixels where |DEM - urban_elev| <=
   current_tolerance
    if count(candidate_pixels) >= min_valid_pixels:
        accept candidate_pixels
        break
```

```
    current_tolerance += step
```

Parameters:

- `initial_tolerance_m`: Starting tolerance (default: 50m, per Raj & Yun (2024))
- `max_tolerance_m`: Maximum tolerance (default: 200m)
- `step_m`: Tolerance increment (default: 25m)
- `min_valid_pixels`: Minimum required pixels (default: 100)

3.4 SUHII Calculation

3.4.1 Primary Metric (Absolute Difference)

The primary output is the scalar SUHII value representing the absolute temperature difference:

$$SUHII = \overline{LST}_{urban} - \overline{LST}_{rural} \tag{7}$$

Calculation Method:

Mean Urban LST ($\overline{LST}_{urban}$): Calculated from all pixels in the urban mask using `numpy.nanmean()` to handle missing data appropriately.

Mean Rural LST ($\overline{LST}_{rural}$): Calculated from all pixels in the rural reference mask that passed the elevation filter.

Units are degrees Celsius (°C). This formulation is used by Raj & Yun (2024) and Fernandes et al. (2024).

3.4.2 Normalized UHI (In-City Method Only)

For the `incity` method, an additional normalized metric is calculated following Ahmad et al. (2024):

$$UHI_{normalized} = \frac{\overline{LST}_{urban} - \overline{LST}_{study}}{\sigma_{study}} \tag{8}$$

Ahmad et al. (2024) report normalized UHI values for Delhi ranging from 8.13 to 10.29 across seasons.

Metric	Units	Interpretation
SUHII (absolute)	°C	"Urban is X degrees warmer than rural"
UHI (normalized)	σ (dimensionless)	"Urban is X standard deviations above mean"

Table 2: SUHII output metrics and their interpretation

3.4.3 Deviation Maps

To visualize spatial heterogeneity, pixel-wise deviation maps are generated, all referenced to the rural mean:

$$Deviation_i = LST_i - \overline{LST}_{rural} \tag{9}$$

Where $Deviation_i$ represents the heat island intensity at pixel location i.

Three spatially-explicit deviation maps are generated:

Output	Spatial Extent	Use Case
`SUHIIUrbanFullDeviation`	Full urban boundary + filtered rural	Visualizing entire urban area
`SUHIIPixelDeviation`	Filtered urban + filtered rural only	Analysis of actual sample pixels
`SUHIIAll`	Urban boundary + entire buffer geometry	Complete analysis domain

Table 3: Output deviation map products

4 Technical Implementation

4.1 Coordinate System Handling

4.1.1 UTM Projection

All spatial operations are performed in a locally-appropriate Universal Transverse Mercator (UTM) projection to ensure metric accuracy for buffer calculations. The tool automatically detects the correct UTM zone based on the city centroid latitude and longitude.

Algorithm:

```
zone = floor((longitude + 180) / 6) + 1
EPSG = 32600 + zone  # Northern hemisphere
EPSG = 32700 + zone  # Southern hemisphere
```

Rationale: Buffer distances specified in meters require a projected coordinate system. UTM provides <0.1% distance distortion within each 6° zone.

4.1.2 Coordinate Flow

```
Input Data (various CRS)
    |
Boundary -> UTM (geometric operations)
Rasters -> UTM (reprojection)
    |
All operations in UTM
    |
Output GeoTIFFs (UTM)
```

4.2 Raster Alignment and Resampling

4.2.1 Grid Alignment

To ensure scientific validity of pixel-wise operations ($LST - \overline{LST}_{rural}$), disparate input datasets (LULC, NDVI, DEM) must be perfectly aligned. All input rasters are aligned to a common grid defined by the LST raster.

Process:

1. LST raster reprojected to UTM (defines master grid)
2. NDVI, DEM, LULC reprojected and resampled to match LST grid using `rasterio.warp.reproject`
3. All arrays share identical dimensions, transform, and CRS

4.2.2 Resampling Methods

The tool supports configurable resampling methods per data layer:

Method	Algorithm	Appropriate For
`nearest`	Nearest neighbor	Categorical data, measured values
`bilinear`	Weighted average of 4 neighbors	Smooth continuous fields

Table 4: Resampling methods and their applications

Scientific Rationale for LST Resampling:

Bilinear interpolation creates weighted average values that were never actually measured. At an urban-rural boundary:

```
Urban pixel (35C) | Rural pixel (28C)
        | Bilinear resampling |
    Artificial 31.5C pixel created
```

This artificially smooths the urban-rural temperature gradient, potentially underestimating SUHII magnitude at boundaries. **Nearest neighbor resampling is recommended for LST** to preserve actual radiometric measurements.

4.3 LST Unit Conversion

The tool supports three input LST formats:

Format	Conversion	Common Sources
`celsius`	None	Pre-processed data
`kelvin`	LST -273.15	MODIS, Landsat Collection 2
`celsius_scaled`	LST / 100	Some GEE exports

Table 5: Supported LST input formats

All internal calculations and outputs are in degrees Celsius.

4.4 Uncertainty Quantification

4.4.1 Standard Error Calculation

The tool reports standard error of the SUHII estimate, assuming urban and rural samples are independent:

$$SE_{SUHII} = \sqrt{SE_{urban}^2 + SE_{rural}^2} \tag{10}$$

Where:

$$SE = \frac{\sigma}{\sqrt{n}} \tag{11}$$

4.4.2 Reported Metrics

Metric	Description
suhii_standard_error	Standard error of SUHII difference
urban_std / rural_std	Sample standard deviations
urban_valid_pixels / rural_valid_pixels	Non-NaN pixel counts

Table 6: Uncertainty metrics reported by the tool

4.4.3 Limitations

The standard error calculation assumes:

- Independent, identically distributed samples
- No spatial autocorrelation

These assumptions are violated in practice (adjacent pixels are correlated). The reported SE should be considered a lower bound; true uncertainty is likely higher.

5 Output Products

5.1 Scalar Results (results.json)

```
{
    "suhii": 3.45,
    "suhii_standard_error": 0.12,
    "urban_mean": 32.50,
    "urban_std": 2.10,
    "urban_elevation_m": 1650,
    "urban_pixels_total": 15000,
    "urban_pixels_valid": 14850,
    "rural_mean": 29.05,
    "rural_std": 1.80,
    "rural_elevation_m": 1680,
    "rural_pixels_total": 8000,
    "rural_pixels_valid": 7950,
    "methodology": {
        "urban_method": "lulc_ndvi",
        "rural_method": "buffer",
        "elevation_tolerance_m": 50.0,
        "min_rural_distance_m": 0.0,
        "lst_input_units": "celsius",
        "resampling": {
            "lst": "nearest",
            "ndvi": "bilinear",
            "dem": "bilinear",
            "lulc": "nearest"
        }
    }
}
```

When `rural_method: "incity"` is used, an additional `normalized_uhi` object is included following Ahmad et al. (2024):

```
{
    "suhii": 3.87,
    "normalized_uhi": {
        "uhi_normalized": 1.42,
```

```
        "study_area_mean": 30.15,
        "study_area_std": 2.72,
        "formula": "(LST_urban - LST_study_area_mean) / LST_study_area_std",
        "citation": "Ahmad, Najar, and Ahmad (2024)",
        "units": "standard deviations"
    }
}
```


Interpretation: A `uhi_normalized` value of 1.42 means the urban mean temperature is 1.42 standard deviations above the study area mean.

5.2 Raster Products

File	Format	Description
`SUHIIUrbanFullDeviation.tif`	GeoTIFF (Float32)	Deviation map, full urban extent
`SUHIIPixelDeviation.tif`	GeoTIFF (Float32)	Deviation map, filtered pixels only
`SUHIIAll.tif`	GeoTIFF (Float32)	Deviation map, entire analysis domain
`aligned_*.tif`	GeoTIFF	Intermediate aligned rasters

Table 7: Raster output products

5.3 Visualization Products

File	Description
`SUHIIUrbanFullDeviation.png`	Deviation map visualization
`SUHIIPixelDeviation.png`	Filtered pixel visualization
`SUHIIAll.png`	Full domain visualization

Table 8: Visualization output products

All visualizations (.png files) use a divergent colormap (RdBu_r) centered on zero deviation from the rural mean.

5.4 Debug Products (optional)

When `outputs.generate_debug_plots: true`:

File	Description
`lulc_check.png`	LULC classification verification
`mask_urban_final.png`	Final urban pixel selection
`mask_rural_final.png`	Final rural pixel selection
`buffer_geometry.png`	Buffer/rural geometry visualization
`final_analysis_area.png`	Combined urban + rural mask
`distance_from_urban.png`	Distance raster (if distance filter used)
`analysis.log`	Detailed processing log

Table 9: Debug output products

6 Validation Considerations

6.1 Comparison with Literature Values

SUHII values should be contextualized against published studies in the study area.

6.2 Ground Validation

Where possible, validate satellite-derived SUHII against:

- Meteorological station air temperature differences
- Vehicle traverse measurements
- Flux tower observations

Raj & Yun (2024) validated MODIS LST against ASOS (Automated Surface Observing System) meteorological stations and found night-time RMSE < 3°C and daytime RMSE > 4°C, attributable to spatial heterogeneity.

6.3 Sensitivity Analysis

Recommended sensitivity tests:

1. **Method comparison:** Run all rural methods, report range
2. **Threshold sensitivity:** Vary NDVI thresholds ± 0.1
3. **Buffer width:** Test 5km, 10km, 15km buffers
4. **Elevation tolerance:** Compare 50m vs. 100m vs. no filter

Raj & Yun (2024) found that method choice matters most for highly urbanized cities (>50% urban land cover), where buffer and incity methods can differ by >1°C.

6.4 Input Data Quality Checks

Users should validate inputs before analysis:

- **Cloud Contamination:** LST rasters should be cloud-masked or taken from timeframes with limited cloud visibility. The tool handles NaN values but cannot detect clouds labeled as valid cold pixels.
- **LULC Accuracy:** If using LULC method, verify the "Built" class ID matches your raster (e.g., Class 6 for Dynamic World).
- **Projection:** Ensure the input city boundary is accurate and has a defined CRS.
- **Temporal Alignment:** Confirm LULC, NDVI and LST acquisition dates are reasonably synchronized.

7 Limitations

7.1 Methodological Limitations

1. **Single-image analysis:** Tool processes one LST image; temporal averaging requires external preprocessing or iterative runs
2. **Administrative boundary dependence:** Results tied to boundary definition
3. **Homogeneity assumption:** Assumes urban/rural categories are internally homogeneous

7.2 Data Limitations

1. **Cloud contamination:** Thermal imagery requires cloud-free conditions

2. **Temporal mismatch:** LULC products may not match LST acquisition date
3. **Resolution effects:** Coarser resolution smooths temperature extremes

7.3 Physical Limitations

1. **Surface vs. air temperature:** SUHII $\neq$ CUHII; interpretation differs
2. **Emissivity effects:** LST retrieval assumes emissivity; urban materials vary
3. **View angle effects:** Off-nadir observations may bias LST

7.4 Elevation Correction Limitations

The current version filters by elevation but does not apply a lapse rate correction factor (e.g., adding $6.5 \times \Delta h$) to the rural temperature. This is a conservative approach (discarding data rather than modifying it).

8 References

Ahmad, B., Najar, M. B., & Ahmad, S. (2024). Analysis of LST, NDVI, and UHI patterns for urban climate using Landsat-9 satellite data in Delhi. *Journal of Atmospheric and Solar-Terrestrial Physics*, 265, 106359. https://doi.org/10.1016/j.jastp.2024.106359

Brown, C. F., Brumby, S. P., Guzder-Williams, B., et al. (2022). Dynamic World, near real-time global 10 m land use land cover mapping. *Scientific Data*, 9(1), 251. https://doi.org/10.1038/s41597-022-01307-4

Fernandes, R., Ferreira, A., Nascimento, V., Freitas, M., & Ometto, J. (2024). Urban heat island assessment in the northeastern state capitals in Brazil using Sentinel-3 SLSTR satellite data. *Sustainability*, 16(11), 4764. https://doi.org/10.3390/su16114764

Keerthi Naidu, B. N., & Chundeli, F. A. (2023). Assessing LULC changes and LST through NDVI and NDBI spatial indicators: A case of Bengaluru, India. *GeoJournal*, 88(4), 4335–4350. https://doi.org/10.1007/s10708-023-10862-1

Kumar, A., Sharma, P., Vyas, N., Sharma, A., Maheshwari, R., & Dutta, P. K. (2025). Land use land cover mapping using random forest and multispectral satellite imagery in Google Earth Engine. *2025 Third International Conference on Networks, Multimedia and Information Technology (NMITCON)*, 1–5. https://doi.org/10.1109/NMITCON65824.2025.11188013

Mentaschi, L., Duveiller, G., Zulian, G., et al. (2022). Global long-term mapping of surface temperature shows intensified intra-city urban heat island extremes. *Global Environmental Change*, 72, 102441. https://doi.org/10.1016/j.gloenvcha.2021.102441

Raj, S., & Yun, G. Y. (2024). Influence of selection of rural reference area for quantifying the surface urban heat islands intensity in major South Korean cities. *Architectural Science Review*, 67(4), 345–356. https://doi.org/10.1080/00038628.2023.2290740

Tempa, K., Ilunga, M., Agarwal, A., & Tashi (2024). Utilizing Sentinel-2 satellite imagery for LULC and NDVI change dynamics for Gelephu, Bhutan. *Applied Sciences*, 14(4), 1578. https://doi.org/10.3390/app14041578

Vignesh, T., Thyagharajan, K. K., Jeyavathana, R. B., & Prasanna Kumar, R. (2022). Land use and land cover classification using Landsat-8 multispectral remote sensing images and long short-term memory-recurrent neural network. AIP Conference Proceedings, 070001. https://doi.org/10.1063/5.0113197

Yang, X., & Yao, L. (2022). Reexamining the relationship between surface urban heat island intensity and annual precipitation: Effects of reference rural land cover. *Urban Climate*, 41, 101074. https://doi.org/10.1016/j.uclim.2021.101074

Zhou, D., Zhao, S., Liu, S., Zhang, L., & Zhu, C. (2014). Surface urban heat island in China's 32 major cities: Spatial patterns and drivers. *Remote Sensing of Environment*, 152, 51–61. https://doi.org/10.1016/j.rse.2014.05.017

A User Guide

A.1 System Requirements

Python: Version 3.11+

OS: Linux, macOS, or Windows

Required Packages:

```
numpy>=1.20
geopandas>=0.10
rasterio>=1.2
scipy>=1.7
matplotlib>=3.4
```

Installation:

```
pip install numpy geopandas rasterio scipy matplotlib
```

Or using the requirements file:

```
pip install -r requirements.txt
```

A.2 Input Data Requirements

Input	Format	Requirements	Notes
LST	GeoTIFF	Single band, any CRS, Float32/Float64	Must be cloud-masked
LULC	GeoTIFF	Integer classes, any CRS	Required for `lulc` methods
NDVI	GeoTIFF	Float $[-1, 1]$, any CRS	Required for all methods
DEM	GeoTIFF	Float (meters), any CRS	Required if elevation correction enabled
Boundary	GeoJSON/Shapefile	Polygon geometry, any CRS	Must encompass urban area

Table 10: Input data requirements

A.3 Quick Start

1. **Prepare input data** in a directory (e.g., `./data/`)
2. **Create configuration file** (`config.json`):

```
{
    "paths": {
        "input_dir": "./data",
```

```
        "output_dir": "./results",
        "lst_file": "LST.tif",
        "lulc_file": "LULC.tif",
        "ndvi_file": "NDVI.tif",
        "dem_file": "DEM.tif",
        "boundary_file": "boundary.geojson"
    },
    "lst_units": "celsius",
    "urban_selection": {
        "method": "lulc_ndvi",
        "urban_classes": [6],
        "water_class": 0,
        "nodata_value": 255,
        "ndvi_max_threshold": 0.3
    },
    "rural_selection": {
        "method": "buffer",
        "buffer_params": {
            "fixed_width_m": 10000
        },
        "vegetation_ndvi_threshold": 0.2,
        "exclude_urban_lulc_classes": true
    },
    "filters": {
        "mask_water": true,
        "use_elevation_correction": true,
        "elevation_params": {
            "initial_tolerance_m": 50,
            "max_tolerance_m": 200,
            "step_m": 25,
            "min_valid_pixels": 100
        }
    },
    "resampling": {
        "lst": "nearest",
        "ndvi": "bilinear",
        "dem": "bilinear"
    },
    "outputs": {
        "generate_debug_plots": true
    }
}
```

3. **Run analysis:**

```
python suhii_tool.py
```

4. **Review outputs** in `./results/`:
   - `results.json` – Quantitative results
   - `SUHIIPixelDeviation.tif` – Deviation map (GeoTIFF)
   - `SUHIIPixelDeviation.png` – Deviation map (visualization)
   - `analysis.log` – Processing log
   - `debug_plots/` – Quality control visualizations

A.4 Generating Config Template

To create a fully-documented configuration template:

```
python suhii_tool.py --generate-config my_config.json
```

B Configuration Reference

B.1 Complete Parameter List

B.1.1 Paths Section

Parameter	Type	Required	Description
`input_dir`	string	Yes	Directory containing input files
`output_dir`	string	Yes	Directory for output products
`lst_file`	string	Yes	LST raster filename
`lulc_file`	string	Conditional	LULC raster filename
`ndvi_file`	string	Yes	NDVI raster filename
`dem_file`	string	Conditional	DEM raster filename
`boundary_file`	string	Yes	Boundary vector filename

Table 11: Path configuration parameters

B.1.2 LST Units

Parameter	Type	Default	Description
`lst_units`	string	`"celsius"`	Input LST unit format: `"celsius"`, `"kelvin"`, `"celsius_scaled"`

Table 12: LST unit configuration

B.1.3 Urban Selection

Parameter	Type	Default	Description
`method`	string	–	`"lulc"`, `"ndvi"`, `"lulc_ndvi"`
`urban_classes`	array[int]	–	LULC class codes for urban
`water_class`	int	0	LULC class code for water
`nodata_value`	int	255	LULC nodata value
`ndvi_max_threshold`	float	0.3	Max NDVI for urban pixels

Table 13: Urban selection configuration parameters

Method Selection Guide:

- `lulc`: Use when you have reliable LULC data and want simple classification
- `ndvi`: Use when LULC unavailable; **WARNING:** may include bare soil
- `lulc_ndvi`: Most rigorous, use when LULC data is not enough to remove vegetated pixels

B.1.4 Rural Selection

Parameter	Type	Default	Description
`method`	string	–	`"buffer"`, `"halo"`, `"three_rings"`, `"incity"`
`fixed_width_m`	float	10000	Buffer width for buffer-/halo methods
`min_distance_from_edge_m`	float	0	Inner exclusion for halo method
`ring_type`	string	`"ua"`	Zone selection for three_rings
`min_rural_distance_m`	float	0	Pixel-level distance filter
`vegetation_ndvi_threshold`	float	0.2	Min NDVI for rural pixels
`exclude_urban_lulc_classes`	bool	true	Exclude urban LULC from rural

Table 14: Rural selection configuration parameters

Method Selection Guide:

- `buffer`: Most common, easy to replicate (Raj & Yun, 2024)
- `halo`: Use for large cities to avoid thermal footprint contamination
- `three_rings`: Best for multi-city comparisons (Fernandes et al., 2024)
- `incity`: Use for intra-urban analysis or isolated cities (Ahmad et al., 2024)

B.1.5 Filters

Parameter	Type	Default	Description
`mask_water`	bool	true	Exclude water bodies from all masks
`use_elevation_correction`	bool	true	Apply elevation filter to rural pixels
`initial_tolerance_m`	float	50	Starting elevation tolerance
`max_tolerance_m`	float	200	Maximum elevation tolerance
`step_m`	float	25	Tolerance increment
`min_valid_pixels`	int	100	Minimum rural pixel count

Table 15: Filter configuration parameters

Elevation Correction Recommendations:

- Flat terrain: Can disable (`use_elevation_correction: false`)
- Moderate terrain: Use defaults ($\pm$50m initial)
- Mountainous: Increase `max_tolerance_m` to 500m

B.1.6 Resampling

Parameter	Type	Default	Description
`lst`	string	`"nearest"`	LST resampling method
`ndvi`	string	`"bilinear"`	NDVI resampling method
`dem`	string	`"bilinear"`	DEM resampling method

Table 16: Resampling configuration parameters

Note: LULC always uses `"nearest"` (categorical data).

Recommendation: Keep LST as `"nearest"` to preserve measured values.

B.2 Example Configurations

B.2.1 Configuration 1: Simple Buffer Method (Recommended for Most Users)

```
{
    "paths": {
        "input_dir": "./data",
        "output_dir": "./results",
        "lst_file": "LST.tif",
        "lulc_file": "LULC.tif",
        "ndvi_file": "NDVI.tif",
        "dem_file": "DEM.tif",
        "boundary_file": "boundary.geojson"
    },
    "lst_units": "kelvin",
    "urban_selection": {
        "method": "lulc_ndvi",
        "urban_classes": [6],
        "water_class": 0,
        "nodata_value": 255,
        "ndvi_max_threshold": 0.3
    },
    "rural_selection": {
        "method": "buffer",
        "buffer_params": {"fixed_width_m": 10000},
        "vegetation_ndvi_threshold": 0.2,
        "exclude_urban_lulc_classes": true
    },
    "filters": {
        "mask_water": true,
        "use_elevation_correction": true,
        "elevation_params": {
            "initial_tolerance_m": 50,
            "max_tolerance_m": 200,
            "step_m": 25,
            "min_valid_pixels": 100
        }
    }
}
```

B.2.2 Configuration 2: In-City Method for Comparative Study

```
{
    "urban_selection": {
        "method": "lulc",
        "urban_classes": [6]
```

```
    },
    "rural_selection": {
        "method": "incity",
        "vegetation_ndvi_threshold": 0.3
    },
    "filters": {
        "mask_water": true,
        "use_elevation_correction": false
    }
}
```

B.2.3 Configuration 3: Three-Rings for Multi-City Comparison

```
{
    "urban_selection": {
        "method": "lulc_ndvi",
        "urban_classes": [6],
        "ndvi_max_threshold": 0.3
    },
    "rural_selection": {
        "method": "three_rings",
        "buffer_params": {"ring_type": "pua"},
        "vegetation_ndvi_threshold": 0.2
    }
}
```

B.2.4 Configuration 4: Mountainous Terrain (Relaxed Elevation Constraints)

```
{
    "filters": {
        "use_elevation_correction": true,
        "elevation_params": {
            "initial_tolerance_m": 50,
            "max_tolerance_m": 500,
            "step_m": 50,
            "min_valid_pixels": 50
        }
    }
}
```

End of Technical Documentation