

Supplementary Material Description for ICLR2026_Paper_20073

The Supplementary materials include:

- A. Dataset description (see this document).
- B. Code usage instructions (see this document).
- C. Code in folder “Code_Paper20073”.

A. Dataset description

MNIST dataset: The MNIST dataset is a seminal dataset in the field of machine learning, particularly for handwritten digit recognition. The dataset is derived from two datasets from NIST: Special Database 1 and Special Database 3, which consist of handwritten digits by NIST employees and American high school students, respectively. The MNIST dataset comprises 70,000 grayscale images of size 28x28 pixels, each representing a handwritten digit from 0 to 9. The dataset is split into two main subsets: a training set containing 60,000 images and a test set with 10,000 images. The images are normalized and centered within a fixed-size box, and they contain grayscale levels due to the anti-aliasing technique used during the normalization process.

The MNIST dataset has been pivotal for training and testing various image processing systems and machine learning models. It serves as a benchmark for evaluating the performance of image classification algorithms. Due to its simplicity and well-structured format, MNIST is often the first choice for training deep learning models. The dataset's images are in a format that is not part of any standard image format, requiring specific programming to read them. However, they can be easily accessed using APIs like PyTorch's torchvision.datasets.MNIST.

Berlin Dataset: The Berlin Dataset is a benchmark dataset for multimodal remote sensing data analysis, particularly for land cover classification tasks. It provides a unique combination of hyperspectral imagery (HSI) and synthetic aperture radar (SAR) data over the same area, which is crucial for advancing the field of remote sensing by enabling the study of complementary information from different sensor modalities. The HSI data in the Berlin Dataset are simulated EnMAP data based on HyMap HS data. The imagery covers an area in Berlin and its surrounding regions, providing detailed spectral information that can be used to distinguish between different materials and objects on the ground. The HSI image is of size 797 x 220 pixels, containing 244 spectral bands within the wavelength range of 400-2500 nm. The SAR data corresponds to the same area as the HSI and is provided by Sentinel-1. It includes dual-polarized SAR data (VV-VH), which offers additional structural and physical information about the scene. The SAR data has a spatial resolution of approximately 13.89 meters and consists of 1723 x 476 pixels. The dataset includes a ground truth generated based on OpenStreetMap data, which is essential for training and validating classification models. The ground reference data helps in accurately assessing the performance of multimodal classification algorithms. Training and test sets containing 2820 and 461851 pixels are provided for this dataset, as shown in Table 1.

Table 1: Berlin dataset with number of training and test samples

No	Class Name	Training Set	Testing Set
1	Forest	443	54511
2	Residential Area	423	268219
3	Industrial Area	499	19067
4	Low Plants	376	58906
5	Soil	331	17095
6	Allotment	280	13025
7	Commercial Area	298	24526
8	Water	170	6502
Total		2820	461851
Percentage		0.61%	99.39%

Houston 2018 Dataset: The Houston 2018 dataset, captured by the Hyperspectral Image Analysis Laboratory and the National Center for Airborne Laser Mapping (NCALM) at the University of Houston, was released for the 2018 IEEE GRSS Data Fusion Contest. It covers the University of Houston campus and the neighboring urban area. The dataset includes hyperspectral data with a spectral range of 380–1050 nm across 48 bands and a spatial resolution of 1 meter. The dataset consists of 4768×1202 pixels, with a training subset size of 2384×601 pixels. Additionally, the LiDAR data is a multispectral image with three bands at 1550 nm, 1064 nm, and 532 nm. This comprehensive dataset represents a challenging urban land-cover and land-use classification task, making it a valuable resource for remote sensing research. The training and testing sets of this dataset are shown in Table 2.

Table 2: Houston2018 dataset with number of training and test samples

No	Class Name	Training Set	Testing Set	Total Set
1	Healthy grass	1000	38196	39196
2	Stressed grass	1000	129008	130008
3	Artificial turf	1000	1736	2736
4	Evergreen trees	1000	53322	54322
5	Deciduous trees	1000	19172	20172
6	Bare earth	1000	17064	18064
7	Water	500	564	1064
8	Residential buildings	1000	157995	158995
9	Non-residential buildings	1000	893769	894769
10	Road	1000	182283	183283
11	Sidewalks	1000	135035	136035
12	Crosswalks	1000	5059	6059
13	Major thoroughfares	1000	184438	185438
14	Highways	1000	38438	39438
15	Railways	1000	26748	27748
16	Paved parking lots	1000	44932	45932
17	Unpaved parking lots	250	337	587
18	Cars	1000	25289	26289
19	Trains	1000	20479	21479
20	Stadium seats	1000	26296	27296
Total		18750	2000160	2018910
Percentage		0.9287%	99.0713%	100%

ISPRS Vaihingen Dataset: The ISPRS Vaihingen Dataset is a high-resolution aerial image dataset that has been widely used for semantic segmentation tasks in remote sensing imagery. It is one of the benchmarks provided by the International Society for Photogrammetry and Remote Sensing (ISPRS) for urban classification, 3D building reconstruction, and semantic labeling. The dataset contains 33 patches of true orthophotos (TOP) and digital surface models (DSM) extracted from a larger mosaic. The ground sampling distance for both the TOP and DSM is 9 cm, providing a high level of detail for analysis. The TOP imagery is provided as 8-bit TIFF files with three bands, corresponding to the near-infrared, red, and green bands captured by the camera. The DSM is a single-band TIFF file with 32-bit floating-point values representing height data. Labeled ground truth is provided for a portion of the data, which is

crucial for training and validating segmentation models. The ground truth data includes six common land cover classes, such as impervious surfaces, buildings, low vegetation, trees, cars, and background

Globe230k dataset: The Globe230k dataset is a benchmark dense-pixel annotation dataset designed for global land cover mapping, created by the Intelligent Mining and Analysis of Remote Sensing big data (IMARS) group at Sun Yat-sen University. This dataset stands out due to its large scale, rich diversity, and multi-modality, making it an invaluable resource for advancing research in land use/land cover (LULC) mapping and semantic segmentation in remote sensing imagery.

The dataset comprises an impressive 232,819 annotated images, each measuring 512x512 pixels with a spatial resolution of 1 meter. It encompasses over 3×10^{10} annotated pixels across 10 first-level categories, providing an extensive dataset for training and testing semantic segmentation models. The images in the Globe230k dataset are sampled from a global perspective, covering an area exceeding 60,000 square kilometers. This ensures a high level of variability and diversity in the dataset, which is crucial for developing models that can generalize well across different geographical regions. Beyond the standard RGB bands, Globe230k includes additional features that are vital for Earth system research. These include the Normalized Differential Vegetation Index (NDVI), Digital Elevation Model (DEM), and dual-polarized Synthetic Aperture Radar (SAR) bands (Vertical-Vertical (VV) and Vertical-Horizontal (VH)). This multi-modal data facilitates research into data fusion and enhances the ability to extract meaningful insights from remote sensing imagery. The dataset has been meticulously annotated by experts and students in surveying and mapping, following a structured annotation pipeline. This ensures a high level of accuracy and consistency in the annotations, which is essential for training robust machine learning models.

The Globe230k dataset has been used to test several state-of-the-art semantic segmentation algorithms, demonstrating its effectiveness in evaluating models across multiple aspects crucial for characterizing land covers, such as multiscale modeling, detail reconstruction, and generalization ability.

IEMOCAP Dataset. IEMOCAP contains dyadic conversation videos between pairs of ten unique speakers. It includes 7,433 utterances and 151 dialogues. Each utterance is annotated with one of six emotion labels: happiness, sadness, neutral, anger, excitement and frustration. The dataset is divided into separate training and testing sets, and the emotion distribution information of IEMOCAP dataset is shown in Table 3.

Table 3: IEMOCAP dataset with number of training and test samples

Class Name	Training Set	Testing Set
Happy	504	144
Sad	839	245
Neutral	1324	384
Angry	933	170
Excited	742	299
Frustrated	1468	381
Total	5810	1623
Percentage	78.16%	21.84%

ImageNet-1k: ImageNet-1k stands out as one of the most influential large-scale image datasets in the field of computer vision. It is part of the broader ImageNet project, which is dedicated to offering a standardized dataset to support visual recognition research. The “1k” indicates the presence of 1,000

object categories, thereby making it a cornerstone for image classification tasks.

Dataset Statistics:

Categories: 1,000 classes (e.g., animals, vehicles, household objects).

Images: Approximately 1.28 million training images, around 50,000 validation images, and roughly 100,000 test images.

Resolution: Images vary in size but are typically high-resolution (e.g., 224x224 or larger following preprocessing).

Source: Images are collected from the web and manually annotated by human labelers.

It is widely utilized to train and evaluate convolutional neural networks (CNNs). Notable models such as AlexNet, ResNet, and Vision Transformers (ViTs) were first validated using ImageNet-1k. Classes are organized within a WordNet hierarchy, facilitating both fine-grained and coarse-grained analysis.

COCO Dataset: The COCO dataset is a cornerstone large-scale dataset in computer vision. Introduced in 2014, it supports research in object detection, instance segmentation, keypoint detection, panoptic segmentation, and image captioning. It has become a standard benchmark for training and evaluating deep learning models in visual recognition tasks, driving advancements in machine perception of complex real-world scenes.

COCO Text-Image Retrieval Extension: This extension of the COCO dataset focuses on cross-modal retrieval, specifically matching text descriptions to images and vice versa. It centers on two core tasks:

1. Retrieving images in response to natural language queries.
2. Generating text captions for images.

Dataset Statistics:

Images: Built on the MS COCO dataset, containing approximately 123,000 images.

Annotations: Each image is paired with 5 human-written captions, totaling approximately 615,000 caption-image pairs.

Vocabulary: Captions use a diverse vocabulary covering objects, actions, and scene contexts (e.g., “a person riding a bicycle on a bridge”).

Key Features: Cross-Modal Nature: Bridges visual and textual modalities, making it essential for image captioning, text-based image search, and multimodal learning. Complex Scenes: Images depict everyday scenes with multiple objects, requiring models to understand spatial relationships and context (e.g., “a dog sitting on a couch next to a window”).

CIFAR-100: CIFAR-100 is a small-scale image dataset designed for fine-grained visual recognition and serves as a more challenging counterpart to CIFAR-10. It is extensively used in academic research to train and test convolutional networks, especially when computational resources are limited.

Dataset Statistics:

Categories: 100 classes, organized into 20 coarse superclasses (e.g., “animals,” “vehicles”) and 100 fine-grained subclasses (e.g., “beaver,” “dalmatian” under “animals”).

Images: 50,000 training images and 10,000 test images, all in 32x32 RGB format.

Content: The images depict everyday objects with significant inter-class similarity (e.g., different species of birds or flowers).

The small image size (32x32) reduces computational overhead, making it ideal for rapid prototyping or lightweight model development.

InternVid Dataset: InternVid is a large-scale video-centric multimodal dataset designed to advance research in video-text representation learning, multimodal understanding, and generation. It addresses the limitations of existing video-language datasets (e.g., low video-text correlation, limited scale/dynamics) by providing high-quality, scalable video-text pairs, enabling the training of transferable video foundation models and supporting diverse downstream tasks.

Dataset Statistics:

Video Quantity: Over 7 million raw videos (sourced from YouTube, excluding videos in publicly available datasets released before April 2023).

Total Duration: Nearly 760,000 hours (average video duration: ~6.4 minutes; 49% of videos <5 minutes, 26% between 5–10 minutes, 8% >20 minutes).

Video Clips: 234 million segmented clips (duration: 2–30+ seconds), generated via scene variance-based trimming (PySceneDetect) to filter static/extreme dynamic content.

Scenes & Actions: Covers 16 scenarios (e.g., People & Blogs, Education, News & Politics, Gaming) and ~6,100 action phrases (derived from ATUS, Kinetics, SomethingSomething, and text corpora).

Resolution: ~85% of videos in 720P; remaining 15% in 360P–720P (prioritizing highest available resolution during collection).

Text Annotations:

Total text volume: 4.1 billion words (clip-level captions generated via multiscale pipeline).

Caption sources: Multiscale generation (coarse-scale: middle-frame captioning with BLIP2; fine-scale: frame-by-frame captioning with Tag2Text + LLM summarization).

Data Diversity: Videos collected from countries with 11 languages (e.g., English, Chinese, Japanese, Russian, French) to ensure cross-lingual representation.

B. Code usage instructions

This section provides instructions for reproducing the experiments presented in the paper. The code is organized to facilitate easy execution and modification for different multimodal learning scenarios.

1. Download and Extract the Supplementary Material

Download the compressed file "Supplementary Material" from the submission system.

Extract the contents to your preferred working directory.

2. Code Structure

After extraction, you will find the main code folder "Code_Paper20073" containing the following Python files:

data_show.py - Data visualization utilities

datadPreprocessing.py- Data preprocessing pipelines

DeepCCASoftHGR_Corr.py- Correlation analysis methods (CCA, Soft-HGR variants)

HGRDecNetworkv_Res50.py - Main network architecture with HGR correlation modules

lion_pytorch.py - Lion optimizer implementation

testHGR.py - Computational efficiency benchmarking

train_hgr_v6.1.py - Main training script

utils.py - Utility functions and helpers

3. Environment Setup

All experiments were implemented using PyTorch 2.1.1. We recommend creating a virtual environment with the following key dependencies:

Python 3.8+

PyTorch 2.1.1

torchvision

NumPy

SciPy

scikit-learn

4. Running Experiments

The framework supports multiple correlation analysis methods. In 'HGRDecNetworkv_Res50.py', you can select from available methods including UniFast HGR, OptFast HGR, DDA, SoftCCA, etc., based on your specific requirements.

4.1 Berlin HSI-SAR Dataset

Download the Berlin dataset and place it in the code directory

Modify lines 15-16 in "train_hgr_v6.1.py":

```
python
```

```
is_Berlin = 1
```

```
is_Houston2018 = 0
```

Execute the training script: python "train_hgr_v6.1.py"

4.2 Houston 2018 HSI-Lidar Dataset

Download the Houston 2018 dataset and place it in the code directory

Modify lines 15-16 in “train_hgr_v6.1.py”:

```
python
```

```
is_Berlin = 0
```

```
is_Houston2018 = 1
```

Execute the training script: python “train_hgr_v6.1.py”

5. Computational Efficiency Benchmarking

To evaluate the computational efficiency of different correlation methods independently of network architecture effects, use the “testHGR.py” script:

This script measures correlation analysis between randomly generated tensors

Modify the batch size parameter (bz) to test different dimensionalities

The script records average execution times for each method across multiple runs

Run with: python “testHGR.py”

6. Customization and Extension

The modular design allows easy integration of new datasets and correlation methods:

Add new datasets by extending the data loading functions in “dataPreprocessing.py”

Implement new correlation methods by following the interface in “DeepCCASoftHGR_Corr.py”

Modify network architectures in “HGRDecNetworkv_Res50.py” while maintaining the correlation module interface

For detailed parameter configurations and additional experimental setups, please refer to the comments within each source file and the main paper methodology section.