Appendix

A. Summary

The appendix is organized as follows:

- § B detailed the training and evaluation settings of our models, including hyper-parameters regarding models and optimizers.
- § C presents a comprehensive introduction on the datasets we use for evaluation and their corresponding metrics.

B. Training Configuration

As shown in Figure 9, we elaborate on the pretraining objectives of MiCo. We provide a more detailed illustration as follows:

B.1. Pretraining Objectives

Omni-modal Contrastive Learning. The omni-modality representations are denoted as $\boldsymbol{z}$. Subsequently, $\boldsymbol{z}$ and $\boldsymbol{z}_T$ are projected into the same space using MLPs. The omni-modal contrastive learning is formulated by the dot product of $\boldsymbol{z}$ and $\boldsymbol{z}_T$. We use v^z and v^T to denote projected vectors:

$$\begin{aligned}\mathcal{L}_{\text{NCE}} = &-\frac{1}{2}\sum_{i=1}^{N_B}\log\frac{\exp(\tau\cdot<v_i^z, v_i^T>)}{\sum_{j=1}^{N_B}\exp(\tau\cdot<v_i^z, v_j^T>)} \\ &-\frac{1}{2}\sum_{i=1}^{N_B}\log\frac{\exp(\tau\cdot<v_i^z, v_i^T>)}{\sum_{j=1}^{N_B}\exp(\tau\cdot<v_j^z, v_i^T>)},\end{aligned} \tag{4}$$

where $<\cdot,\cdot>$, N_B and τ denote the dot product, batch size, and a learnable parameter.

Omni-modal Feature Matching Process is designed to improve the semantic alignment between multimodal (knowledge modalities) and textual features. We employ an MLP layer to perform binary predictions p_v of $(\boldsymbol{z}, \boldsymbol{z}_T)$. Following a hard negative mining strategy[50], we assigns $y = 1$ if features are matched, and $y = 0$ otherwise.

$$\mathcal{L}_{\text{logits}} = \mathbb{E}_{(v_i^z, v_i^T)\sim(\mathcal{Z},\mathcal{T})}\left[y\log p_v + (1-y)\log\left(1-p_v\right)\right] \tag{5}$$

Omni-modal Caption Generation Process. We employ conditional causal masked (60%) language modeling for generative omni-modal reasoning. In specific, a single-directional causal attention mask is used to avoid information leakage, and the masked tokens are reconstructed using a prediction layer of BERT [16]. We use c_m and $c_{<m}$ to denote masked tokens and former tokens, respectively.

$$\mathcal{L}_{\text{causal}} = -\mathbb{E}_{(v_i^T, v_i^T)\sim(\mathcal{V},\mathcal{T})}\log P\left(c_m \mid c_{<m}, v^z\right) \tag{6}$$

B.2. Pretraining Settings

We detail the specific pretraining configurations of MiCo, focusing on the multi-dataset joint training corpora, the dataset mix ratios for each corpus, and the learning objectives for each corpus. To improve data quality, we employed a trained vision captioner to generate new captions for the CC4M datasets, replacing the original captions. Although MiCo has only been trained for 300,000 steps, it has already demonstrated outstanding performance on various downstream tasks. We anticipate that further increasing the number of training steps will significantly enhance the model's capabilities.

The pretraining of MiCo involves a combination of different datasets, each contributing uniquely to the model's learning process. With a parameter size of 1.0 billion and a sample size of 334 million, the model utilizes a diverse training corpus to achieve its results.

1. VAST-27M: This dataset contributes 324 million samples to the training process. With a batch size of 2048, the model undergoes 160,000 steps, completing one epoch.

2. VALOR-1M: In this dataset, 1 million samples are used with a batch size of 1024. The training spans 70,000 steps, which equates to approximately 71.7 epochs.

3. WavCaps, CC4M, and WebVid-2.5M: These datasets are combined, contributing 9 million samples in total. The batch size for this combined dataset is 1024, and the model is trained over 70,000 steps, resulting in 8.0 epochs.

The careful selection and combination of these datasets, along with the application of new, high-quality captions for the CC4M datasets, enhance the training efficiency and the quality of the learned representations.

B.3. Fine-tuning Settings

We detail the downstream task finetuning settings, specifying the learning rate, batch size, epoch, training objectives, and resolution. The configurations also include the number of sampled video frames or audio clips used in training and testing phases. Here are the comprehensive settings:

Retrieval Tasks (RET)

- **Image-Text Modality**
  - **MSCOCO**: Learning rate of 1e-5, batch size of 256, 5 epochs, with the objective for retrieval, and a resolution of 384.
  - **Flickr**: Learning rate of 1e-5, batch size of 256, 5 epochs, with the objective for retrieval, and a resolution of 384.
- **Audio-Text Modality (A-T)**
  - **ClothoV1/V2**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for retrieval, using 3 audio clips during both training and testing.
  - **AudioCaps**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for retrieval, using 1 audio clip during both training and testing.

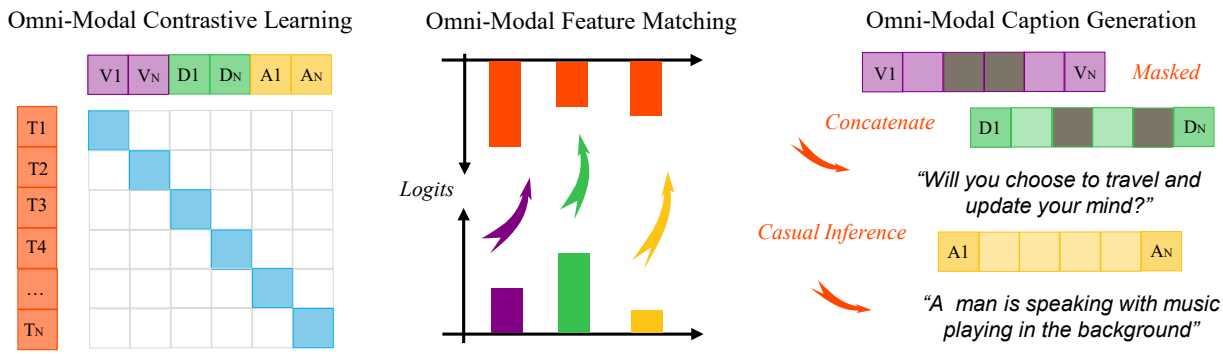

Figure 9. Pretraining Objectives of MiCo. It contains 3 parts of loss functions: contrastive learning with InfoNCE loss, feature matching with the logits entropy, and causal caption generation.

- **Multi-modal (MM)**
  - **MSRVTT**: Learning rate of 2e-5, batch size of 64, 3.6 epochs, with the objective for retrieval, using 8 video frames during training and 16 during testing, with a resolution of 224.
  - **YouCook2**: Learning rate of 3e-5, batch size of 64, 30 epochs, with the objective for retrieval, using 8 video frames during training and 16 during testing, with a resolution of 224.
  - **VALOR-32K**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for retrieval, using 8 video frames during both training and testing, with a resolution of 224.
  - **VATEX**: Learning rate of 2e-5, batch size of 64, 2.5 epochs, with the objective for retrieval, using 8 video frames during training and 16 during testing, with a resolution of 224.
  - **DiDeMo**: Learning rate of 2e-5, batch size of 64, 40 epochs, with the objective for retrieval, using 8 video frames during training and 32 during testing, and 2 audio clips during both training and testing, with a resolution of 224.
  - **ANET**: Learning rate of 2e-5, batch size of 64, 20 epochs, with the objective for retrieval, using 8 video frames during training and 32 during testing, and 2 audio clips during both training and testing, with a resolution of 224.

Captioning Tasks (CAP)

- **Image-Text Modality**
  - **MSCOCO**: Learning rate of 1e-5, batch size of 64, 5 epochs, with the objective for caption, and a resolution of 480.
  - **MSCOCO(SCST)**: Learning rate of 2.5e-6, batch size of 64, 2.5 epochs, with the objective for caption, and a resolution of 480.
- **Audio-Text Modality (A-T)**
  - **ClothoV1/V2**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for caption, using 3 audio clips during both training and testing.
  - **AudioCaps**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for caption, using 1 audio clip during both training and testing.
- **Multi-modal (MM)**
  - **MSRVTT**: Learning rate of 2e-5, batch size of 128, 10 epochs, with the objective for caption, using 8 video frames during both training and testing, with a resolution of 224.
  - **YouCook2**: Learning rate of 3e-5, batch size of 64, 30 epochs, with the objective for caption, using 8 video frames during training and 16 during testing, with a resolution of 224.
  - **VALOR-32K**: Learning rate of 1e-5, batch size of 64, 10 epochs, with the objective for caption, using 8 video frames during training and 12 during testing, with a resolution of 224.

Question Answering Tasks (QA)

- **Visual-Text Modality (Vis)**
  - **MSVD-QA**: Learning rate of 1e-5, batch size of 64, 10 epochs, with the objective for QA, using 8 video frames during training and 14 during testing, with a resolution of 224.
  - **TGIF-FrameQA**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for QA, using 4 video frames during both training and testing, with a resolution of 224.
  - **VQAv2**: Learning rate of 2e-5, batch size of 128, 20 epochs, with the objective for QA, and a resolution of 384.
- **Multi-modal (MM)**
  - **MSRVTT-QA**: Learning rate of 2e-5, batch size of 64, 4.5 epochs, with the objective for QA, using 8 video frames and 1 audio clip during both training and testing, with a resolution of 224.
  - **MUSIC-AVQA**: Learning rate of 2e-5, batch size of

Algorithm 1 Multimodal Context Pretraining Algorithm, PyTorch-like

```
def train(video_pixels=None, image_pixels=None,
    depth_pixels=None, audio_spectrograms=None):
   # Get Mixed Data
   modal_inputs = [video_pixels, image_pixels,
       depth_pixels, audio_spectrograms]
   modal_captions = [video_captions, image_captions
       , depth_captions, audio_captions]

   # Extract Features
   modal_feats = [self.encoder(modal) for modal in
       modal_inputs if modal is not None]
   multimodal_feats = torch.cat(modal_feats)
   concatenated_captions = ''.join(modal_captions)
   text_feats = self.text_encoder(
       concatenated_captions)

   # Losses
   contra_loss = Contrasive_Loss(multimodal_feats,
       text_feats)
   matching_loss = Matching_Loss(modal_captions,
       multimodal_feats)
   gen_loss = Generation_Loss(modal_captions.mask
       (0.6), multimodal_feats)

   # Total Loss
   loss = contra_loss + matching_loss + gen_loss

return loss
```

64, 20 epochs, with the objective for QA, using 8 video frames and 2 audio clips during both training and testing, with a resolution of 224.

– **ANET-QA**: Learning rate of 2e-5, batch size of 64, 10 epochs, with the objective for QA, using 8 video frames during training and 16 during testing, and 2 audio clips during both training and testing, with a resolution of 224.

These settings have been optimized to balance efficiency and performance, even though most hyper-parameters are not precisely tuned.

For evaluation purposes, we employ different strategies tailored to specific tasks:

1. Retrieval Tasks: All candidates are initially ranked using Omni-modal Contrastive Loss. Following this, the Top-50 candidates undergo a reranking process through the Omni-modal Matching Process.

2. Captioning Tasks: Beam search with a beam size of 3 is utilized to generate captions, ensuring a comprehensive exploration of possible outputs.

3. Question Answering (QA) Tasks: These are treated as open-ended generative problems. Questions are used as prefixes, and answers are generated without any constraints, allowing for flexible and contextually appropriate responses.

For comparisons with state-of-the-art (SOTA) models and ablation studies, we use the following evaluation metrics: 1) Retrieval Tasks: Recall@1. 2) Captioning Tasks: CIDEr. 3) QA Tasks: Accuracy (Acc) These metrics provide a comprehensive assessment of the model's performance across different types of tasks.

C. Datasets and Metrics

Dataset Split To split the mix of datasets into subsets of 1M, 10M, 110M, and 334M video clips while preserving their diversity and quality, we employed a proportional stratified sampling method. Initially, the dataset, which spans over 15 categories (including music, gaming, education, entertainment, and animals) and includes vision, audio, depth, normal maps, and text modalities, was organized and labeled. Stratified random sampling was then used to ensure each subset accurately reflected the distribution of categories and modalities present in the full dataset. This method involved selecting samples proportionally from each category to maintain representative distributions. The vision and audio captions were also kept proportional in length and quantity, ensuring that each subset retained the comprehensive characteristics of the original dataset.

C.1. Single-modality Evaluation Details

Text. The MMLU (Massive Multitask Language Understanding) benchmark is designed to evaluate the multitask accuracy of language models across 57 diverse tasks, including subjects like mathematics, history, and biology. It assesses models' abilities to generalize and apply knowledge in various domains, providing a comprehensive measure of text understanding and reasoning skills.

Image. We conduct experiments on ImageNet-1K [14], a dataset comprising approximately 1.3 million images across 1,000 categories. In line with common practices [17, 65, 66, 89], base-scale models are trained for 300 epochs. Large-scale models undergo pre-training on ImageNet-22K, which includes 14.2 million images, for 90 epochs, followed by fine-tuning on ImageNet-1K for an additional 20 epochs.

Thermal and Hyperspectral data understanding. We conduct experiments on infrared image recognition using the RegDB dataset, X-ray scan analysis with the Chest X-Ray dataset [79], and hyperspectral data recognition using the Indian Pine dataset[2].

Depth. The NYU Depth Dataset (NYU-D) comprises RGB and depth image pairs captured from indoor scenes. It

[2] https://github.com/danfenghong/IEEE_TGRS_SpectralFormer/blob/main/data/IndianPine.mat

Table 10. Detailed training configurations of MiCo for multimodal learning. Apart from the configurations shown in the table, for image tasks, we use random left-right flipping, random resized crop, color jitter of 0.4, Auto-augment, and no repeated augmentation for every model.

settings	Image		Audio		Video		Depth & Normal Map	
	ViT-L	ViT-g	ViT-L	ViT-g	ViT-L	ViT-g	ViT-L	ViT-g
Input Shape	224	224	224	224	224	224	224	224
batch size	4096	512	4096	512	4096	512	4096	512
optimizer	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW
LR	4×10^{-3}	5×10^{-5}	4×10^{-3}	5×10^{-5}	4×10^{-3}	5×10^{-5}	4×10^{-3}	5×10^{-5}
LR schedule	cosine	cosine	cosine	cosine	cosine	cosine	cosine	cosine
weight decay	0.05	1×10^{-8}	0.05	1×10^{-8}	0.05	1×10^{-8}	0.05	1×10^{-8}
warmup epochs	5	0	5	0	5	0	5	0
epochs	90	30	90	30	90	20	90	20
mixup alpha	0.8	0.0	0.8	0.0	0.8	0.0	0.8	0.0
cutmix alpha	1.0	0.0	1.0	0.0	1.0	0.0	1.0	0.0
erasing prob.	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25
dropout rate	0.1	0.2	0.1	0.2	0.1	0.3	0.2	0.3

Algorithm 2 Dataset Split Algorithm

```
import pandas as pd
from sklearn.model_selection import train_test_split

# Assume ‘data‘ is a DataFrame containing the full dataset with columns [’category’, ’vision_caption’, ’
    audio_caption’, ’depth’, ’normal’, ’subtitle’]
# Adding an ’index’ column to keep track of the original indices
data[’index’] = data.index

# Define the sizes of each subset
subset_sizes = [1e6, 1e7, 1.1e7, 3.34e7]

# Function to create stratified samples
def create_subset(data, size):
   subset, _ = train_test_split(data, train_size=size, stratify=data[’category’], random_state=42)
   return subset

# Creating subsets
subset_1M = create_subset(data, 1e6)
subset_10M = create_subset(data, 1e7)
subset_110M = create_subset(data, 1.1e7)
subset_334M = create_subset(data, 3.34e7)

# Reset index for each subset
subset_1M.reset_index(drop=True, inplace=True)
subset_10M.reset_index(drop=True, inplace=True)
subset_110M.reset_index(drop=True, inplace=True)
subset_334M.reset_index(drop=True, inplace=True)
```

includes 1,449 densely labeled pairs for training and testing, along with over 400,000 unlabeled frames.

Audio. For audio recognition, Audioset-2M dataset comprises over 2 million human-labeled 10-second audio clips drawn from YouTube videos. It covers a wide range of 527 sound event classes, providing a comprehensive resource for training and evaluating audio event detection and classification models.

Video. The Kinetics-700 dataset contains 700,000 video clips covering 700 human action classes, used for action recognition tasks. The MSR-VTT dataset includes 10,000 video clips paired with multiple textual descriptions, sup-

porting video captioning, retrieval, and content understanding research.

Time-series. Global Weather Forecasting [95] includes global, regional, and Olympics data from NCEI and CMA, comprising hourly weather measurements from thousands of stations. Evaluation involved splitting data into training, validation, and test sets (7:1:2) using MSE and MAE metrics.

Graph. PCQM4M-LSC dataset is a large-scale collection of 4.4 million organic molecules, each with up to 23 heavy atoms and associated quantum-mechanical properties. Aimed at predicting molecular properties through machine learning, this dataset is highly relevant for applications in drug discovery and material science.

Tabular. The fraud dataset comprises transaction records, including features like transaction amount, location, time, and user information. It is designed for machine learning models to detect fraudulent activities. This dataset is crucial for developing and testing algorithms to enhance security in financial systems and reduce economic losses due to fraud.

IMU. The Ego4D dataset includes inertial measurement unit (IMU) data captured from wearable devices, providing detailed motion and orientation information. This dataset supports research in human activity recognition, augmented reality, and robotics, offering comprehensive insights into human movements and interactions with the environment.

C.2. Cross-modality Evaluation Details

We evaluated MiCo across several well-known downstream datasets, including MSRVTT, VATEX, YouCook2, VALOR-32K, MSVD, DiDeMo, ActivityNet Caption, TGIF, MUSIC-AVQA, Clotho, AudioCaps, MSCOCO, Flickr30K, and VQAv2. The specific train/validation/test splits for these benchmarks are detailed below:

Retrieval Tasks

Audio-Text Modality (A-T)

- **ClothoV1** [19]: This dataset includes 2,893 audio clips for training and 1,045 for validation. The corresponding captions number 14,465 for training and 5,225 for validation.
- **ClothoV2** [19]: Contains 3,839 audio clips for training and 1,045 for validation, with 19,195 captions for training and 5,225 for validation.
- **AudioCaps** [41]: Comprises 49,291 audio clips for training, 428 for validation, and 816 for testing, along with 49,291 captions for training, 2,140 for validation, and 4,080 for testing.

Video-Text Modality (V-T)

- **MSRVTT** [98]: Comprises 10K video clips and 200K captions, spanning diverse topics such as human activities, sports, and natural landscapes. We evaluate text-to-video retrieval, video captioning, and video QA using this dataset. Contains 9,000 videos for training and 1,000 for testing, with 180,000 captions for training and 1,000 for testing.
- **YouCook2** [115]: Comprises 14K video clips extracted from 2K instructional cooking videos on YouTube. Each video features multiple actions performed by chefs, along with corresponding textual descriptions and temporal annotations. Includes 10,337 videos for training and 3,492 for validation, with matching captions.
- **VALOR-32K** [9]: An audiovisual video-language benchmark containing 32K 10-second video clips sourced from AudioSet [25]. Each clip includes annotations with captions that describe both the visual and audio content. Consists of 25,000 videos for training, 3,500 for validation, and 3,500 for testing, each with corresponding captions.
- **DiDeMo** [4]: Comprises 10K long-form videos sourced from Flickr, with each video annotated with four short sentences in temporal order. For this benchmark, we concatenate these short sentences and evaluate 'paragraph-to-video' retrieval, using the official split. Features 8,394 videos for training, 1,065 for validation, and 1,003 for testing, along with their captions.
- **ActivityNet (ANET)** [43]: Includes 20K long-form videos (average length of 180 seconds) from YouTube, accompanied by 100K captions. We evaluate text-to-video retrieval and video QA on this dataset. Comprises 10,009 videos for training and 4,917 for testing, with corresponding captions.
- **LSMDC** [80]: Contains 101,046 videos for training, 7,408 for validation, and 1,000 for testing, with corresponding captions.

Captioning Tasks

Audio-Text Modality (A-T)

- **ClothoV1** [19]: This dataset includes 2,893 audio clips for training and 1,045 for validation. The corresponding captions number 14,465 for training and 5,225 for validation.
- **ClothoV2** [19]: Contains 3,839 audio clips for training and 1,045 for validation, with 19,195 captions for training and 5,225 for validation.
- **AudioCaps** [41]: Comprises 49,838 audio clips for training, 495 for validation, and 975 for testing, along with 49,438 captions for training, 2,475 for validation, and 4,875 for testing.

Video-Text Modality (V-T)

- **MSRVTT** [98]: Contains 6,513 videos for training, 497 for validation, and 2,990 for testing, with 130,260 captions for training, 9,940 for validation, and 59,800 for testing.
- **YouCook2** [115]: Includes 10,337 videos for training and 3,492 for validation, with matching captions.
- **VALOR-32K** [9]: Consists of 25,000 videos for training, 3,500 for validation, and 3,500 for testing, each with corresponding captions.
- **VATEX** [91]: Consists of 41,250 video clips sourced from the Kinetics-600 dataset [40], accompanied by 825,000 sentence-level descriptions. Contains 25,991 videos for training, 3,000 for validation, and 6,000 for testing, with 259,910 captions for training, 30,000 for validation, and 60,000 for testing.

Question Answering (QA) Tasks

Video-Text Modality (V-T)

- **MSRVTT-QA** [96]: Contains 6,513 videos for training, 497 for validation, and 2,990 for testing, with 158,581 QA pairs for training, 12,278 for validation, and 72,821 for testing.
- **MUSIC-AVQA** [48]: An audiovisual video QA benchmark containing over 45K Q-A pairs, covering 33 different question templates across various modalities and question types. Includes 9,277 videos for training, 3,815 for validation, and 6,399 for testing, with 32,087 QA pairs for training, 4,595 for validation, and 9,185 for testing.
- **ANET-QA** [106]: Comprises 3,200 videos for training, 1,800 for validation, and 800 for testing, with 32,000 QA pairs for training, 18,000 for validation, and 8,000 for testing.

Image-Based Tasks

- **MSCOCO** [59]: Comprises 123K images, each paired with 5 annotated captions. We evaluate text-to-image retrieval and image captioning on this dataset.
- **Flickr30K** [75]: Contains 31K images, each paired with five descriptive captions. This dataset is widely used for evaluating image captioning and text-to-image retrieval tasks.

Visual Question Answering

- **VQAv2** [28]: A large-scale Visual Question Answering dataset comprising over 265K images and 1.1M questions, designed to improve the balance of answer types per question. This dataset is used to evaluate models' abilities to understand and reason about visual content by providing accurate answers to questions based on the images.