JOINT CLASS–TIME LEARNING FOR VIDEO CLASSIFICATION WITH MULTI-INSTANCE PARTIAL-LABEL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-instance partial-label learning (MIPL) addresses inexact supervision in both the instance and label spaces, which can be applied to video classification. However, bag-level labels do not explicitly supervise the correspondence between candidate classes and temporal evidence. We propose PIVOTMIPL, which couples label disambiguation with temporal evidence allocation through a joint class–time assignment. Occupancy-regularized spherical matching associates contextualized video features while learning nonuniform temporal mass and discouraging excessive concentration. During training, candidate-restricted inference recomputes the assignment within the candidate label set. A dual-marginal KL projection then constructs a structured teacher that incorporates momentum-refined class beliefs while preserving the proposal's temporal occupancy. A single plan-level KL objective aligns the full-space predictor with this teacher. Our analysis characterizes when candidate re-solving differs from masking and shows that, under the stated construction, the joint objective decomposes into class-marginal and class-conditional temporal supervision. We construct VCMIPL benchmarks from Breakfast, DoTA, and FineAction using model-generated candidate labels and evaluate the method across four feature representations. Extensive experimental results demonstrate that PIVOTMIPL outperforms existing MIPL algorithms in both effectiveness and efficiency. Code is available at `https://anonymous.4open.science/r/PIVOTMIPL-202B/`.

1 INTRODUCTION

Learning to classify videos from weak annotations requires identifying both the true label and its supporting temporal evidence. Video-level labels do not specify which frames or clips support the target class within a sequence that may also contain background, transitions, and related activities, resulting in inexact supervision in the input space (Zhang et al., 2025b). At the class level, candidate pseudolabeling retains several plausible classes from model predictions to generate the candidate label set (Zhang et al., 2024). It contains the true label and several false-positive labels, which does not specify the true label and provides inexact supervision in the output space (Wang et al., 2025). Multi-instance partial-label learning (MIPL) captures both forms of inexact supervision by representing each example as a bag of instances paired with a candidate label set containing its true label and several false-positive labels (Tang et al., 2024c). We formulate this setting as *video classification with MIPL (VCMIPL)*, where each bag is an ordered sequence of frame or clip features and its supervision is a candidate label set.

Existing methods offer several ways to learn from inexact supervision. Attention-based MIPL couples bag representation learning with momentum-based label disambiguation (Tang et al., 2023). In partial-label learning, prototype-guided disambiguation (Wang et al., 2022b; Jin et al., 2025) and transport-based label refinement (Wang et al., 2022a; Gu et al., 2025) exploit feature geometry and assignment constraints, respectively. Weakly-supervised video classification jointly learns classification and temporal instance selection from explicit video-level labels (Wang et al., 2017). However, in VCMIPL, temporal evidence must be inferred while the true class remains ambiguous. Although instance selection and label disambiguation can interact during learning, refining bag-level class probabilities alone does not determine how temporal evidence is allocated to each candidate class.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Distinct class–time assignments can yield identical class probabilities, even when they assign the same total mass to every temporal position. This motivates learning from a structured class–time target that combines refined class beliefs with temporal allocation inferred within the candidate set.

In this paper, we propose a VCMIPL method named PIVOTMIPL, i.e., *Plan Inference via Variable-Occupancy Transport for Multi-Instance Partial-Label learning*, which represents each video through a class–prototype–time assignment. A temporal encoder produces contextualized features that are matched to multiple prototypes per class on the unit sphere. Temporal occupancy, the mass assigned to each position, is inferred jointly with the assignment, with regularization discouraging excessive concentration. During training, we re-solve the assignment within the candidate label set to obtain a proposal and refine its class marginal through momentum accumulation. A KL projection then constructs the closest teacher to this proposal subject to the refined class marginal and the proposal's temporal occupancy. A single plan-level KL objective aligns the full-space predictor with the structured teacher. At test time, prediction uses only the full-space assignment, requiring neither candidate labels nor teacher construction.

We construct three VCMIPL benchmarks based on Breakfast (Kuehne et al., 2014), DoTA (Yao et al., 2023), and FineAction (Liu et al., 2022) datasets, using four feature representations. Extensive experimental results on the benchmarks show that PIVOTMIPL achieves the highest mean accuracy and balanced accuracy in 10 of 12 configurations, averaging gains of 3.16% and 3.26% over the strongest baseline per configuration, respectively. And the accuracy is significantly higher than the strongest baseline in 8 of 12 configurations. Ablations support the contributions of occupancy regularization, adaptive temporal allocation, joint class–time supervision, dual-marginal teacher construction, multiple prototypes, and temporal encoding. Gains persist in 10 configurations against AGOPMIPL (Cao et al., 2026), which is the strongest baseline, with the same temporal encoder. Further analyses show favorable accuracy–training-time trade-offs in several configurations and stable performance over the tested regularization ranges on three representative configurations.

Our contributions are threefold: First, we introduce a structured framework for VCMIPL that couples candidate-restricted temporal allocation with label disambiguation through a joint assignment and a teacher constrained by class and temporal marginals. Second, we analyze the inference and supervision mechanisms, characterizing the relationship between candidate re-solving and masking, the concentration control induced by occupancy regularization, and the effective supervision provided by the joint objective. Third, we construct three VCMIPL benchmarks with model-generated candidate label sets with four feature representations, and evaluate the framework against representative MIPL methods on the benchmarks.

2 RELATED WORK

2.1 MULTI-INSTANCE PARTIAL-LABEL LEARNING

MIPL learns from bags with ambiguous candidate labels, addressing inexact supervision in both instance and label spaces. Tang et al. (2024c) introduced instance-level probabilistic disambiguation, whereas Tang et al. (2023) used attention-based bag representations and momentum-refined label beliefs. Subsequent methods exploit complementary candidate- and non-candidate-label information (Tang et al., 2024b), attention and prediction margins (Tang et al., 2024a), probabilistic generative modeling of latent label distributions (Yang et al., 2024), and instance- and bag-level dependencies and heterogeneity for computational efficiency (Yang et al., 2025). Recent approaches incorporate candidate-aware propensity scoring and deconfounding (Luo et al., 2026), feature and label propagation over bag- and instance-level graphs (Zhu et al., 2026), feature geometry based on average gradient outer products (Cao et al., 2026), aggregation uncertainty through evidential learning (Jie et al., 2026), and prediction calibration (Tang et al., 2026). However, our focus is on the specific coupling of candidate-restricted temporal occupancy, historical class beliefs, and a teacher constrained by class and temporal marginals.

2.2 WEAKLY SUPERVISED VIDEO LEARNING

Weakly supervised video learning infers temporal evidence from video-level labels. UntrimmedNets represents videos as temporal segments and jointly learns segment selection and video-level classification (Wang et al., 2017). Later methods model intra- and inter-segment dependencies (He et al.,

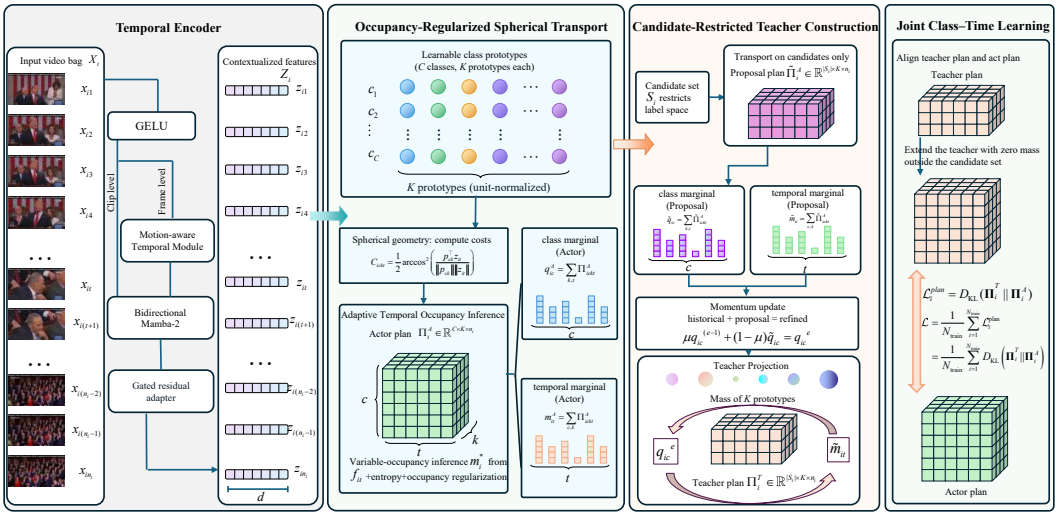

Figure 1: The framework of PIVOTMIPL

2022), organize evidence into proposals (Ren et al., 2023), and infer salient snippet features from variations between neighboring time steps (Yun et al., 2024). PseudoFormer progressively converts weak predictions into more reliable pseudo supervision for temporal localization (Liu & Liu, 2025), while multimodal large language models provide semantic priors (Zhang et al., 2025a). These methods generally assume unambiguous video labels. Our setting additionally requires identifying the true class within a candidate set and linking it to supporting temporal evidence.

3 METHODOLOGY

3.1 PROBLEM FORMULATION AND OVERVIEW

Let $\mathcal{D} = \{(\mathbf{X}_i, S_i)\}_{i=1}^{N_{\text{train}}}$ denote a MIPL training dataset of N_{train} videos. Each video is represented as an ordered multi-instance bag $\mathbf{X}_i = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{in_i}) \in \mathbb{R}^{n_i \times d}$, where n_i is the number of instances and d is the feature dimension. Each video has a single target label $Y_i \in \mathcal{Y} = \{1, \ldots, C\}$, but is annotated only with a nonempty candidate label set $S_i \subseteq \mathcal{Y}$ containing Y_i. We adopt collective bag classification, allowing multiple temporal positions to jointly support the target class without requiring every raw frame to belong to that class.

The framework of PIVOTMIPL is shown in Figure 1. A feature-aware temporal encoder produces contextualized instance representations. Occupancy-regularized spherical transport matches these representations to multiple prototypes per class, yielding an actor plan over the full label space with adaptive temporal occupancy while discouraging excessive concentration. During training, we re-solve the transport problem within the candidate label set to obtain a proposal and refine its class marginal through momentum accumulation. We then construct a structured teacher plan that matches the refined class marginal while preserving the proposal's temporal occupancy. A plan-level KL objective aligns the actor with the detached teacher, jointly supervising class assignments and class-conditional temporal distributions.

3.2 TEMPORAL ENCODER

The temporal encoder maps a multi-instance bag $\mathbf{X}_i \in \mathbb{R}^{n_i \times d}$ to contextualized features $\mathbf{Z}_i \in \mathbb{R}^{n_i \times d}$. A local difference module processes frame-level ResNet (He et al., 2016) and DINOv3 (Siméoni et al., 2025) features, while clip-level SlowFast (Feichtenhofer et al., 2019) and VideoMAEv2 (Wang et al., 2023) features bypass this module because they already represent local clips. Both paths then use bidirectional Mamba-2 (Dao & Gu, 2024) and a gated residual adapter. We denote row-wise affine maps by $\mathrm{Aff}_a(H) = HW_a^\top + \mathbf{1}b_a^\top$, with learnable weights and biases,

and apply layer normalization (LN) over the feature dimension. The input projection maps the features from d-dimensional to d_h-dimensional channels:

$$\mathbf{H}_i^{(0)} = \mathrm{GELU}\big(\mathrm{Aff}_{\mathrm{in}}(\mathrm{LN}(\mathbf{X}_i))\big), \qquad \mathbf{H}_i^{(0)} \in \mathbb{R}^{n_i \times d_h}. \tag{1}$$

Here $\mathrm{Aff}_{\mathrm{in}}$ maps d-dimensional input channels to d_h-dimensional hidden channels.

Local Temporal Differences. For frame-level features, we first smooth the projected sequence with a three-position moving average, then compute backward and forward differences:

$$\widehat{\mathbf{h}}_{it}^{(0)} = \frac{1}{3}\left(\mathbf{h}_{i,t-1}^{(0)} + \mathbf{h}_{it}^{(0)} + \mathbf{h}_{i,t+1}^{(0)}\right), \qquad \Delta_{it}^{-} = \widehat{\mathbf{h}}_{it}^{(0)} - \widehat{\mathbf{h}}_{i,t-1}^{(0)}, \qquad \Delta_{it}^{+} = \widehat{\mathbf{h}}_{i,t+1}^{(0)} - \widehat{\mathbf{h}}_{it}^{(0)}. \tag{2}$$

Here $\mathbf{h}_{it}^{(0)}$ denotes the t-th row of $\mathbf{H}_i^{(0)}$. The moving average uses replicate padding at both sequence boundaries. We set $\Delta_{i1}^{-} = \mathbf{0}$ and $\Delta_{in_i}^{+} = \mathbf{0}$; when $n_i = 1$, both difference vectors are zero. We concatenate the two difference sequences along the channel dimension, denoted by $[\cdot;\cdot]$, project them from $2d_h$-dimensional channels to d_h-dimensional ones, and apply two convolutional branches:

$$\mathbf{D}_i = \mathrm{Aff}_{\Delta}([\boldsymbol{\Delta}_i^{-}; \boldsymbol{\Delta}_i^{+}]), \qquad \mathbf{U}_i^{(j)} = \mathrm{SiLU}\big(\mathcal{C}_{\delta_j}(\mathbf{D}_i)\big), \qquad (\delta_1, \delta_2) = (1, 2). \tag{3}$$

$\mathcal{C}_{\delta_j}$ contains a depthwise temporal convolution with kernel size of 5 and dilation δ_j, followed by a 1×1 pointwise convolution for channel mixing. The two branches have receptive fields of 5 and 9 positions over $\mathbf{D}_i$, respectively, and each produces an output of shape $n_i \times d_h$. Here $\mathrm{SiLU}(x) = x\sigma(x)$, where σ is the sigmoid function. We fuse the branch outputs and form a gated motion residual:

$$\begin{aligned} \mathbf{M}_i &= \mathrm{SiLU}\big(\mathrm{LN}(\mathrm{Aff}_M([\mathbf{U}_i^{(1)}; \mathbf{U}_i^{(2)}]))\big), \\ \mathbf{G}_i &= \sigma\big(\mathrm{Aff}_g([\mathbf{H}_i^{(0)}; \mathbf{M}_i])\big), \qquad \mathcal{M}(\mathbf{H}_i^{(0)}) = \mathbf{H}_i^{(0)} + \mathrm{Dropout}_{0.1}(\mathbf{G}_i \odot \mathbf{M}_i). \end{aligned} \tag{4}$$

Both Aff_M and Aff_g map $2d_h$ channels to d_h. The gate $\mathbf{G}_i$ weights the input-dependent motion representation $\mathbf{M}_i$ at each temporal position and hidden channel. The input to the bidirectional encoder is $\widetilde{\mathbf{H}}_i^{(0)} = \mathcal{M}(\mathbf{H}_i^{(0)})$ for frame-level features and $\widetilde{\mathbf{H}}_i^{(0)} = \mathbf{H}_i^{(0)}$ for clip-level features.

Bidirectional Temporal Context. We apply L bidirectional Mamba-2 blocks in sequence:

$$\mathbf{H}_i^{(L)} = \mathcal{B}^{(L)} \circ \cdots \circ \mathcal{B}^{(1)}(\widetilde{\mathbf{H}}_i^{(0)}). \tag{5}$$

Each block processes the sequence in forward and reverse temporal order. The backward output is restored to the original order, and the two outputs are combined using a feature-dependent directional gate before a feed-forward residual update. The output adapter projects the hidden features back to d-dimensional channels and adds a gated residual to the input:

$$\mathbf{Z}_i = \mathbf{X}_i + \sigma(\gamma)\,\mathrm{Dropout}\big(\mathrm{Aff}_{\mathrm{out}}(\mathrm{LN}(\mathbf{H}_i^{(L)}))\big). \tag{6}$$

Here $\mathrm{Aff}_{\mathrm{out}}$ maps d_h-dimensional channels to d-dimensional ones, and $\gamma \in \mathbb{R}$ is a learnable scalar. The identity path preserves the input features, while the residual incorporates temporal context. Therefore, the output positions are contextualized features rather than independent raw snippets.

3.3 Occupancy-Regularized Spherical Transport

Spherical Matching And Occupancy Regularization. Each class has K learnable prototypes $\mathbf{p}_{ck} \in \mathbb{R}^d$. For nonzero features and prototypes, define the unit-normalized vectors $\bar{\mathbf{z}}_{it} = \mathbf{z}_{it}/\|\mathbf{z}_{it}\|_2$ and $\bar{\mathbf{p}}_{ck} = \mathbf{p}_{ck}/\|\mathbf{p}_{ck}\|_2$. The matching cost is half the squared geodesic distance on the unit sphere:

$$C_{ickt} = \tfrac{1}{2}\arccos^2(\bar{\mathbf{p}}_{ck}^{\top}\bar{\mathbf{z}}_{it}). \tag{7}$$

For any nonempty admissible class set $\mathcal{A} \subseteq \mathcal{Y}$, define $\Delta_i^{\mathcal{A}}$ as the unit simplex over $(c,k,t) \in \mathcal{A} \times \{1,\dots,K\} \times \{1,\dots,n_i\}$. With temporal occupancy $m_{it} = \sum_{c\in\mathcal{A},k} \Pi_{ickt}$, our inference operator is

$$\begin{aligned} \mathcal{T}_{\varepsilon,\lambda}(\mathbf{C}_i; \mathcal{A}) = \arg\min_{\boldsymbol{\Pi}\in\Delta_i^{\mathcal{A}}} \Bigg\{ &\sum_{c\in\mathcal{A},k,t} \Pi_{ickt} C_{ickt} + \varepsilon \sum_{c\in\mathcal{A},k,t} \Pi_{ickt}(\log \Pi_{ickt} - 1) \\ &+ \frac{\lambda}{2}\left(n_i \sum_t m_{it}^2 - 1\right)\Bigg\}, \end{aligned} \tag{8}$$

where $\varepsilon > 0$ and $\lambda \geq 0$. Here, "transport" refers to a regularized assignment with freely optimized class and temporal marginals. Let $\mathbf{u}_i$ be the uniform temporal distribution, with $u_{it} = 1/n_i$. The occupancy penalty can be written as

$$R_{\mathrm{occ}}(\mathbf{m}_i) = \chi^2(\mathbf{m}_i \| \mathbf{u}_i) = e^{D_2(\mathbf{m}_i \| \mathbf{u}_i)} - 1 = n_i \sum_t m_{it}^2 - 1, \tag{9}$$

where χ^2 and D_2 denote the Pearson χ^2 divergence and the order-2 Rényi divergence, respectively. This penalty discourages excessive mass on a few temporal positions without enforcing uniform occupancy. It controls concentration but does not identify action locations. More information about the concentration properties can be found in Appendix A.5.

Factorized Inference. For fixed temporal occupancy, the conditional minimization over classes and prototypes admits a Gibbs solution. Define

$$\mathcal{Z}_{it}^{\mathcal{A}} = \sum_{c \in \mathcal{A}, k} e^{-C_{ickt}/\varepsilon}, \qquad \rho_{ickt}^{\mathcal{A}} = \frac{e^{-C_{ickt}/\varepsilon}}{\mathcal{Z}_{it}^{\mathcal{A}}}, \qquad f_{it}^{\mathcal{A}} = -\varepsilon \log \mathcal{Z}_{it}^{\mathcal{A}}. \tag{10}$$

Substituting $\Pi_{ickt} = m_{it} \rho_{ickt}^{\mathcal{A}}$ into Eq. (8) and dropping terms independent of $\mathbf{m}_i$ gives

$$\mathbf{m}_i^\star = \arg\min_{\mathbf{m}_i \in \Delta_{n_i}} \left\{ \langle \mathbf{m}_i, \mathbf{f}_i^{\mathcal{A}} \rangle + \varepsilon D_{\mathrm{KL}}(\mathbf{m}_i \| \mathbf{u}_i) + \frac{\lambda}{2} \left(n_i \|\mathbf{m}_i\|_2^2 - 1 \right) \right\}, \tag{11}$$

where Δ_{n_i} is the unit simplex in $\mathbb{R}^{n_i}$. The remaining problem is to optimize an n_i-dimensional occupancy vector, with its normalization multiplier determined via scalar root-finding. The optimal plan, full-space actor plan, and actor class marginal are, respectively,

$$\Pi_{ickt}^\star = m_{it}^\star \rho_{ickt}^{\mathcal{A}}, \qquad \mathbf{\Pi}_i^A = \mathcal{T}_{\varepsilon,\lambda}(\mathbf{C}_i; \mathcal{Y}), \qquad q_{ic}^A = \sum_{k,t} \Pi_{ickt}^A. \tag{12}$$

Occupancy Solver. For $\lambda > 0$, the occupancy can be expressed via the normalization multiplier ξ_i:

$$m_{it}(\xi_i) = \frac{\varepsilon}{\lambda n_i} W\left(\frac{\lambda}{\varepsilon} e^{(\xi_i - f_{it}^{\mathcal{A}})/\varepsilon} \right), \qquad g_i(\xi_i) = \sum_t m_{it}(\xi_i) - 1, \tag{13}$$

where W is the principal Lambert-W function, defined by $W(x)e^{W(x)} = x$. We choose the multiplier to satisfy $g_i(\xi_i) = 0$. The described solver initializes $\xi_i^{(0)} = -\varepsilon \log \sum_t u_{it} \exp(-f_{it}^{\mathcal{A}}/\varepsilon)$ and applies Newton updates:

$$\xi_i^{(r+1)} = \xi_i^{(r)} - \frac{g_i(\xi_i^{(r)})}{\displaystyle\sum_t \frac{m_{it}(\xi_i^{(r)})}{\varepsilon + \lambda n_i m_{it}(\xi_i^{(r)})}}. \tag{14}$$

When $\lambda = 0$, the solution instead takes the form $m_{it}^\star = \exp(-f_{it}^{\mathcal{A}}/\varepsilon) / \sum_s \exp(-f_{is}^{\mathcal{A}}/\varepsilon)$. This case requires a separate branch as Eq. (13) contains $1/\lambda$, and entropy regularization remains active. Appendix A.1 derives these expressions and establishes the uniqueness of the positive optimum.

We compute $f_{it}^{\mathcal{A}}$ using log-sum-exp and evaluate the conditional assignments in the log domain. For $\lambda > 0$, the Lambert-W term is evaluated directly from its log argument: letting $h_{it} = \log(\lambda/\varepsilon) + (\xi_i - f_{it}^{\mathcal{A}})/\varepsilon$, we solve $w_{it} + \log w_{it} = h_{it}$ for $w_{it} = W(e^{h_{it}})$ by Newton iteration, avoiding explicit construction of $e^{h_{it}}$. We then update the scalar normalization multiplier by Newton iteration and recover $m_{it} = \varepsilon w_{it}/(\lambda n_i)$. The occupancy solve is performed in double precision, and we monitor both its normalization and KKT stationarity residuals. For $\lambda = 0$, we directly use the closed-form softmax solution.

Class Belief Update. During training, we construct a proposal on the candidate label set S_i using the full-space actor's current costs and temperature:

$$\widetilde{\mathbf{\Pi}}_i = \mathcal{T}_{\varepsilon,\lambda}(\mathrm{sg}(\mathbf{C}_i); S_i), \qquad \widetilde{q}_{ic} = \sum_{k,t} \widetilde{\Pi}_{ickt}, \qquad \widetilde{m}_{it} = \sum_{c \in S_i, k} \widetilde{\Pi}_{ickt}. \tag{15}$$

Here, sg denotes stop-gradient. We recompute the partition functions and free energies over candidate classes and apply the same occupancy solver to obtain $\widetilde{\mathbf{m}}_i$. Therefore, this restriction determines class support and can change temporal allocation.

Re-solving generally differs from globally masking and renormalizing the actor. Global masking also changes temporal mass, but it need not optimize the restricted occupancy-regularized objective. Appendix A.2 gives the precise equivalence conditions, including the $\lambda = 0$ boundary. The method uses re-solving rather than a fixed-time conditional renormalization. Initialize $q_{ic}^{(0)} = \mathbf{1}[c \in S_i]/|S_i|$. At epoch e, combine the stored belief with the current proposal marginal:

$$q_{ic}^{(e)} = \mu q_{ic}^{(e-1)} + (1-\mu)\widetilde{q}_{ic}, \qquad \mu \in [0,1),\ c \in S_i, \tag{16}$$

and set $q_{ic}^{(e)} = 0$ for $c \notin S_i$. The EMA averages class marginals, not full plans or network parameters. Each training bag is visited once per epoch. At its visit in epoch e, the stored $q_i^{(e-1)}$ is combined with the current detached proposal marginal to form $q_i^{(e)}$; this refined belief is used immediately for teacher construction and is stored after the optimizer step for the next epoch.

3.4 Candidate-Restricted Teacher Construction

The teacher incorporates historical class mass while retaining the proposal's current temporal occupancy. We obtain it through the KL projection on the candidate support:

$$\begin{aligned} \mathbf{\Pi}_i^T &= \arg\min_{\mathbf{\Pi} \geq 0} D_{\mathrm{KL}}(\mathbf{\Pi} \| \widetilde{\mathbf{\Pi}}_i), \\ \text{subject to} \quad \sum_{k,t} \Pi_{ickt} &= q_{ic}^{(e)} \quad (c \in S_i), \qquad \sum_{c \in S_i, k} \Pi_{ickt} = \widetilde{m}_{it} \quad (t = 1, \ldots, n_i). \end{aligned} \tag{17}$$

Let $\widetilde{M}_{ict} = \sum_k \widetilde{\Pi}_{ickt}$ denote the proposal's class–time matrix. The teacher has the multiplicative form $\Pi_{ickt}^T = \alpha_{ic}\widetilde{\Pi}_{ickt}\beta_{it}$. Starting from positive scaling factors, we alternate the updates

$$\alpha_{ic} \leftarrow \frac{q_{ic}^{(e)}}{\sum_t \widetilde{M}_{ict}\beta_{it}}, \qquad \beta_{it} \leftarrow \frac{\widetilde{m}_{it}}{\sum_{c \in S_i} \widetilde{M}_{ict}\alpha_{ic}}. \tag{18}$$

Each update enforces one target marginal; both constraints hold at convergence. We perform the marginal scaling in the log domain using log-sum-exp denominators. Computing the scaling factors requires only the $|S_i| \times n_i$ class–time matrix. The full teacher is then recovered by multiplying each proposal entry by $\alpha_{ic}\beta_{it}$, with the same factors applied across prototypes.

We extend the teacher with zeros outside S_i to obtain $\widehat{\mathbf{\Pi}}_i^T$. The training objective is

$$\mathcal{L} = \frac{1}{N_{\text{train}}} \sum_i \mathcal{L}_i^{\text{plan}}, \qquad \mathcal{L}_i^{\text{plan}} = D_{\mathrm{KL}}\left(\mathrm{sg}(\widehat{\mathbf{\Pi}}_i^T) \| \mathbf{\Pi}_i^A\right). \tag{19}$$

Under the same-cost construction and exact teacher projection, the prototype-conditional distributions of the actor and teacher coincide on the candidate support. Therefore, the plan-level objective reduces to

$$\mathcal{L}_i^{\text{plan}} = D_{\mathrm{KL}}(\mathbf{q}_i^T \| \mathbf{q}_i^A) + \sum\nolimits_{c: q_{ic}^T > 0} q_{ic}^T D_{\mathrm{KL}}\big(P_i^T(t \mid c) \| P_i^A(t \mid c)\big), \tag{20}$$

where q_{ic}^T and q_{ic}^A are the teacher and actor class marginals, respectively, and $P_i^T(t \mid c)$ and $P_i^A(t \mid c)$ denote their class-conditional temporal distributions obtained after marginalizing over the prototype dimension. Thus, beyond class-marginal alignment, the joint objective explicitly supervises the class-conditional temporal allocation. A complete derivation and the corresponding current-point gradient equivalence are provided in Appendix A.3.

The loss uses the full-space actor without renormalizing it over S_i, thereby penalizing mass assigned to non-candidate classes. Detaching the entire target-construction pipeline gives

$$\nabla_\theta \mathcal{L}_i^{\text{plan}} = -\sum\nolimits_{c \in S_i, k, t} \Pi_{ickt}^T \nabla_\theta \log \Pi_{ickt}^A. \tag{21}$$

The actor gradient propagates through both temporal occupancy and Gibbs conditional assignments to the matching costs, prototypes, and temporal encoder. We differentiate through the unrolled actor solver, and no gradients pass through proposal inference, EMA updates, or teacher projection.

Algorithm 1: PIVOTMIPL training and prediction

Initialize parameters and $q_{ic}^{(0)} = \mathbf{1}[c \in S_i]/|S_i|$.
For each epoch e and each training bag (X_i, S_i):
  Encode valid ordered positions and compute $\mathbf{C}_i$.
  Solve full-space actor $\mathbf{\Pi}_i^A = \mathcal{T}(\mathbf{C}_i; \mathcal{Y})$.
  Solve proposal $\widetilde{\mathbf{\Pi}}_i = \mathcal{T}(\mathrm{sg}(\mathbf{C}_i); S_i)$.
  Update class belief using Eq. (16).
  Scale the proposal to $(\mathbf{q}_i^{(e)}, \widetilde{\mathbf{m}}_i)$ using Eq. (18).
  Extend teacher by zero outside S_i and detach it.
  Backpropagate the actor KL, clip the global gradient norm, step the optimizer, and store the detached refined belief.
At test time, return $\arg\max_c \sum_{k,t} \Pi_{ickt}^A$ without candidates.

The resulting supervision acts on class–time associations parameterized by multiple prototypes. Under the stated same-cost construction, the actor and teacher have identical conditional distributions over prototypes given class and time on the candidate support. The prototype-conditional KL term is therefore zero, although prototypes still affect the class–time distribution and can receive gradients through it. The full KL decomposition, the corresponding gradient equivalence at the teacher construction point, and boundary cases are established in Appendix A.3–A.4.

3.5 JOINT CLASS–TIME LEARNING AND PREDICTION

Algorithm 1 summarizes the per-bag training procedure. Training uses one complete video bag per optimization step: we compute one plan-level loss, perform one backward pass and gradient-clipped optimizer update, and then write the detached refined class belief back to the training dataset. Since bags are processed individually, no temporal padding is introduced. At test time, we compute only $\mathbf{Z}_i$, the matching costs, and the full-space actor, and predict $\hat{y}_i = \arg\max_{c \in \mathcal{Y}} \sum_{k,t} \Pi_{ickt}^A$.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmark Construction We construct three VCMIPL benchmarks, i.e., Breakfast-MIPL, DoTA-MIPL, and FineAction-MIPL, from the Breakfast (Kuehne et al., 2014), DoTA (Yao et al., 2023), and FineAction (Liu et al., 2022) datasets, respectively. For FineAction, we retain single-label videos and exclude classes with fewer than 50 videos. The resulting subsets contain 1,989 Breakfast videos (10 classes), 4,577 DoTA videos (9 classes), and 10,675 FineAction videos (78 classes). Appendix B reports candidate-set and feature statistics. We use ResNet (He et al., 2016) and DINOv3 (Siméoni et al., 2025) for frame-level features, and SlowFast (Feichtenhofer et al., 2019) and VideoMAEv2 (Wang et al., 2023) for clip-level features. To reduce the annotation cost, we generate candidate labels using a VideoMAEv2 recognizer equipped with a TimeSformerHead in MMAction2 (MMAction2 Contributors, 2020). We fine-tune the recognizer on four folds and generate predictions for the held-out fold, repeating this procedure for all five folds. Fine-tuning uses 30 epochs on DoTA and 100 epochs on both Breakfast and FineAction. Following Zhang et al. (2024), we first select labels for each video in descending order of confidence until their cumulative confidence reaches 95%. A selected label is retained only if the video also ranks among the top 50% of videos by confidence for that label.

Comparative Methods We compare PIVOTMIPL with seven MIPL baselines: DEMIPL (Tang et al., 2023), ELIMIPL (Tang et al., 2024b), MIPLMA (Tang et al., 2024a), PROMIPL (Yang et al., 2024), PSMIPL (Luo et al., 2026), DUALG (Zhu et al., 2026), and AGOPMIPL (Cao et al., 2026). We exclude MIPLGP (Tang et al., 2024c), FASTMIPL (Yang et al., 2025), and UAMIPL (Jie et al., 2026) because their released implementations require all instance-level features to be loaded into memory, making them impractical for these video benchmarks under the available memory budget.

Table 1: Classification accuracy on the benchmark datasets. Results are reported as (mean ± std %). The best result in each column is bolded, and the second-best results are underlined. ●/○ indicates whether PIVOTMIPL is statistically superior/inferior to the corresponding method under a two-sided paired t-test at $p < 0.05$.

Dataset	Method	DINOv3	VideoMAEv2	ResNet	SlowFast
Breakfast -MIPL	DEMIPL	11.4% ± 0.4% ●	25.7% ± 10.0% ●	10.1% ± 0.7% ●	11.0% ± 0.5% ●
	ELIMIPL	27.8% ± 0.8% ●	52.2% ± 1.9% ●	26.1% ± 1.7% ●	11.3% ± 0.1% ●
	MIPLMA	33.9% ± 1.5% ●	63.6% ± 2.1% ●	30.5% ± 2.4% ●	11.3% ± 0.1% ●
	PROMIPL	62.2% ± 1.2% ●	83.8% ± 3.4% ●	43.7% ± 1.2% ●	41.6% ± 1.1% ●
	PSMIPL	47.3% ± 5.0% ●	70.3% ± 7.3% ●	11.0% ± 1.4% ●	33.0% ± 0.6% ●
	DUALG	45.7% ± 3.2% ●	64.9% ± 4.4% ●	34.3% ± 2.0% ●	14.4% ± 1.6% ●
	AGOPMIPL	46.3% ± 4.2% ●	**87.6%** ± 4.3%	33.3% ± 3.1% ●	50.4% ± 2.5% ●
	PIVOTMIPL	**66.3%** ± 1.0%	86.1% ± 4.5%	**52.3%** ± 3.4%	**55.9%** ± 1.8%
DoTA -MIPL	DEMIPL	45.6% ± 1.3% ●	68.1% ± 0.7% ●	42.6% ± 0.4% ●	41.2% ± 1.6% ●
	ELIMIPL	45.8% ± 1.6% ●	68.3% ± 1.0% ●	43.4% ± 0.5% ●	41.7% ± 1.0% ●
	MIPLMA	46.8% ± 1.1% ●	69.2% ± 0.9% ●	43.7% ± 1.0% ●	41.8% ± 1.1% ●
	PROMIPL	49.2% ± 1.8% ●	72.5% ± 1.5%	45.9% ± 2.2% ●	48.0% ± 0.4%
	PSMIPL	49.8% ± 0.9% ●	74.1% ± 0.9%	44.6% ± 1.3% ●	26.0% ± 13.0% ●
	DUALG	46.1% ± 2.6% ●	69.6% ± 0.8% ●	44.3% ± 1.2% ●	43.6% ± 0.9% ●
	AGOPMIPL	49.7% ± 1.3%	69.9% ± 1.8% ●	45.2% ± 0.7% ●	47.0% ± 0.1% ●
	PIVOTMIPL	**52.7%** ± 2.0%	**74.7%** ± 1.8%	**49.6%** ± 1.2%	**48.8%** ± 0.6%
FineAction -MIPL	DEMIPL	41.5% ± 1.6% ●	52.1% ± 1.6% ●	21.7% ± 1.8% ●	50.2% ± 2.1% ●
	ELIMIPL	42.0% ± 1.0% ●	49.6% ± 2.7% ●	22.9% ± 2.0% ●	49.3% ± 1.7% ●
	MIPLMA	55.4% ± 1.4% ●	68.3% ± 0.5% ●	35.7% ± 3.0% ●	57.2% ± 2.3% ●
	PROMIPL	80.7% ± 1.5% ●	92.0% ± 3.4%	69.7% ± 1.4% ●	82.4% ± 2.2% ●
	PSMIPL	60.5% ± 1.4% ●	66.2% ± 1.5% ●	43.0% ± 2.0% ●	32.7% ± 14.9% ●
	DUALG	58.1% ± 1.4% ●	74.2% ± 0.8% ●	46.4% ± 1.3% ●	63.9% ± 0.8% ●
	AGOPMIPL	82.2% ± 1.3% ●	**96.3%** ± 0.5% ○	71.1% ± 1.0% ●	84.2% ± 2.2% ●
	PIVOTMIPL	**88.5%** ± 1.0%	94.0% ± 1.4%	**76.2%** ± 1.2%	**88.3%** ± 1.0%

Implementation Details PIVOTMIPL is implemented in PyTorch and trained for 40 epochs on an NVIDIA H100 GPU using AdamW with $(\beta_1, \beta_2) = (0.9, 0.999)$ and zero weight decay. The learning rates are 5×10^{-3} for the prototypes and 5×10^{-4} for the temporal encoder. We evaluate PIVOTMIPL over the same five predefined folds used for cross-fitting the candidate-label recognizer. We report classification accuracy (ACC) and balanced accuracy (BACC) as mean ± standard deviation over five folds, where BACC is the mean recall over ground-truth classes represented in each test fold. More implementation details are provided in Appendix B.3.

4.2 MAIN EXPERIMENTAL RESULTS

Table 1 shows that PIVOTMIPL achieves the highest mean ACC in 10 of the 12 dataset–feature configurations, with an average gain of 3.16% over the strongest baseline in each configuration. Beyond the overall improvement, the performance pattern supports the central motivation of our framework. Most competing MIPL methods ultimately resolve ambiguity through bag-level class supervision, without explicitly retaining which temporal evidence supports each candidate class. In contrast, PIVOTMIPL preserves this class–time association throughout disambiguation, so that the plan-level objective constrains both the class belief and its supporting temporal allocation. This distinction is particularly evident with the frame-level DINOv3 and ResNet representations, where PIVOTMIPL consistently ranks first across all three datasets. These representations leave temporal dependencies largely unresolved before MIPL learning, making explicit evidence allocation especially useful for separating informative instances from background or class-irrelevant content. The main exceptions occur in VideoMAEv2, where strong baselines already achieve highly competitive performance. A plausible explanation is that its clip-level representations already encode substantial local spatiotemporal context, reducing the remaining ambiguity that can be resolved by downstream temporal restructuring. Nevertheless, PIVOTMIPL is inferior to AGOPMIPL in only one configuration while being significantly superior to the strongest competing method in eight, suggesting that its advantage primarily stems from coupling label disambiguation with temporal evidence allocation rather than from a particular feature backbone.

Table 2: Ablation study of PIVOTMIPL.

Dataset	Feature	PIVOTMIPL	A1	A2	A3	A4	A5	A6
Breakfast -MIPL	DINOv3	66.4%	62.6%	**72.1%**	66.1%	64.8%	64.3%	57.1%
	VideoMAEv2	77.4%	**77.8%**	77.1%	70.3%	67.1%	72.6%	75.6%
	ResNet	52.2%	42.6%	**54.4%**	51.4%	50.3%	47.9%	35.4%
	SlowFast	**53.9%**	48.4%	53.4%	52.6%	51.6%	53.6%	49.9%
DoTA -MIPL	DINOv3	**54.3%**	50.3%	51.7%	53.5%	52.6%	51.3%	49.3%
	VideoMAEv2	72.8%	72.0%	70.9%	72.2%	72.7%	**73.0%**	72.7%
	ResNet	**50.9%**	48.1%	49.8%	50.5%	50.4%	47.2%	47.5%
	SlowFast	**48.4%**	46.4%	47.8%	45.9%	47.1%	46.5%	43.9%
FineAction -MIPL	DINOv3	**90.0%**	89.7%	86.3%	79.0%	86.3%	85.7%	89.4%
	VideoMAEv2	92.4%	93.7%	96.3%	88.7%	89.1%	91.1%	**96.7%**
	ResNet	**78.1%**	75.5%	71.9%	72.6%	77.6%	75.4%	75.8%
	SlowFast	**90.0%**	80.7%	86.5%	84.9%	86.9%	88.0%	64.3%

4.3 ABLATION STUDY

We evaluate six ablation variants across all 12 dataset–feature configurations using Fold 1 ACC at epoch 40. Table 2 presents the individual results. Moreover, a summary of the ablations across twelve dataset–feature configurations is provided in Appendix C.3.1. **A1** removes occupancy regularization by setting $\lambda = 0$, while retaining entropy regularization and learned temporal occupancy. **A2** fixes temporal occupancy to the uniform distribution, $m_{it} = 1/n_i$. **A3** replaces the joint KL objective with a class-level KL objective. **A4** replaces the dual-marginal teacher projection with class-marginal-only rescaling, removing the constraint that preserves the proposal's temporal occupancy. **A5** uses a single prototype per class, i.e., $K = 1$. **A6** removes the temporal encoder.

Table 2 shows that every ablation reduces mean ACC across the 12 configurations. Notably, A1 and A2 reduce ACC in 10 and 9 configurations, favoring regularized yet adaptive temporal allocation. A3 and A4 degrade performance across all configurations, even with explicit class-time supervision and occupancy-preserving teacher construction. A5 and A6 each reduce ACC in 11 configurations, supporting multiple prototypes and temporal contextualization. Temporal encoding has the largest average impact, while the joint objective and dual-marginal teacher deliver the most consistent gains.

Appendix C.1 reports the BACC results, with PIVOTMIPL achieving the highest mean performance in 10 of the 12 configurations. To assess the role of temporal encoding, In Appendix C.3.2, we compare PIVOTMIPL with AGOPMIPL, for which we adopt the same temporal encoder as PIVOTMIPL. PIVOTMIPL outperforms this baseline in 10 of the 12 configurations, suggesting that the encoder alone does not account for the observed gains. Appendix C.2 reports favorable trade-offs between accuracy and computational efficiency across most evaluated settings. Appendix C.4 shows limited performance variation over the tested ranges of the main transport hyperparameters.

5 CONCLUSION

We presented PIVOTMIPL, a structured framework coupling candidate-label disambiguation with temporal evidence allocation for video multi-instance partial-label learning. Occupancy-regularized spherical transport matches contextualized features to multiple prototypes per class, learning nonuniform temporal assignments while discouraging excessive concentration. Candidate-restricted inference and dual-marginal KL projection construct a teacher that integrates momentum-refined class beliefs while preserving the proposal's temporal occupancy. A single plan-level KL objective aligns the full-space predictor with this teacher. Our analysis characterizes when candidate re-solving differs from masking and establishes, under the proposed construction, the objective's decomposition into class-marginal alignment and class-conditional temporal supervision. Across Breakfast-MIPL, DoTA-MIPL, and FineAction-MIPL with four feature representations, PIVOTMIPL achieves the highest mean ACC and BACC in 10 of 12 configurations. Ablations support the joint objective and dual-marginal teacher, while encoder-matched comparisons suggest gains beyond temporal encoding. These findings support resolving label ambiguity through explicit class–time associations rather than class probabilities alone.

6 STATEMENTS

AI USE STATEMENT

Large Language Models (LLMs) were used to aid in the writing and polishing of the manuscript. Specifically, we used an LLM to assist in refining the language, improving readability, and ensuring clarity in various sections of the paper. The model helped with tasks such as sentence rephrasing, grammar checking, and enhancing the overall flow of the text.

It is important to note that the LLM was not involved in the ideation, research methodology, or experimental design. All research concepts, ideas, and analyses were developed and conducted by the authors. The LLM's contributions focused solely on improving the paper's linguistic quality, with no involvement in the scientific content or data analysis.

The authors take full responsibility for the content of the manuscript, including any text generated or polished by the LLM. We have ensured that the LLM-generated text adheres to ethical guidelines and does not contribute to plagiarism or scientific misconduct.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. In this study, no human subjects or animal experimentation was involved. All datasets used, including Breakfast-MIPL, DoTA-MIPL and FineAction-MIPL, were sourced in compliance with relevant usage guidelines, ensuring no violation of privacy. We have taken care to avoid any biases or discriminatory outcomes in our research process. No personally identifiable information was used, and no experiments were conducted that could raise privacy or security concerns. We are committed to maintaining transparency and integrity throughout the research process.

REPRODUCIBILITY STATEMENT

We have made every effort to ensure that the results presented in this paper are reproducible. All code and datasets have been made publicly available in an anonymous repository to facilitate replication and verification. The experimental setup, including training steps, model configurations, and hardware details, is described in detail in the paper. We have also provided a full description of PIVOTMIPL, to assist others in reproducing our experiments.

Additionally, VCMIPL datasets such as Breakfast-MIPL, DoTA-MIPL and FineAction-MIPL are publicly available, ensuring consistent and reproducible evaluation results.

We believe these measures will enable other researchers to reproduce our work and further advance the field.

REFERENCES

Nan Cao, Xu Zhao, and Teng Zhang. Enhanced multi-instance partial label learning via average gradient outer product. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 10041–10071, 2024.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. SlowFast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6201–6210, 2019.

Yiyang Gu, Binqi Chen, Zihao Chen, Ziyue Qiao, Xiao Luo, Junyu Luo, Zhiping Xiao, Wei Ju, and Ming Zhang. MATE: Masked optimal transport with dynamic selection for partial label graph learning. *Artificial Intelligence*, 348:104396, 2025.

Bo He, Xitong Yang, Le Kang, Zhiyu Cheng, Xin Zhou, and Abhinav Shrivastava. ASM-Loc: Action-aware segment modeling for weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13915–13925, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Gaowen Jie, Fumiao Wang, Gaojie Song, Luojun Lin, Yuanlong Yu, and Qinghai Zheng. Uncertainty-aware multi-instance partial-label learning via evidential deep model. *Neurocomputing*, 701:134578, 2026.

Yu-Jie Jin, Ya-Sha Wang, and Xu Chu. NPC: Negative prototypical contrasting for label disambiguation of partial label learning. *Journal of Computer Science and Technology*, 40(5):1386–1400, 2025.

Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 780–787, 2014.

Yi Liu, Limin Wang, Yali Wang, Xiao Ma, and Yu Qiao. FineAction: A fine-grained video dataset for temporal action localization. *IEEE Transactions on Image Processing*, 31:6937–6950, 2022.

Ziyi Liu and Yangcen Liu. Bridge the gap: From weak to full supervision for temporal action localization with PseudoFormer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8711–8720, 2025.

Hanjie Luo, Yu-Xuan Zhang, Zhengchun Zhou, Weisha Liu, and Ming Li. Propensity scoring for multi-instance partial-label learning. In *Web and Big Data (APWeb-WAIM 2025)*, pp. 239–255, 2026.

MMAction2 Contributors. OpenMMLab's next generation video understanding toolbox and benchmark. `https://github.com/open-mmlab/mmaction2`, 2020.

Huan Ren, Wenfei Yang, Tianzhu Zhang, and Yongdong Zhang. Proposal-based multiple instance learning for weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2394–2404, 2023.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025.

Wei Tang, Weijia Zhang, and Min-Ling Zhang. Disambiguated attention embedding for multi-instance partial-label learning. In *Advances in Neural Information Processing Systems 36*, pp. 56756–56771, 2023.

Wei Tang, Yin-Fang Yang, Zhaofei Wang, Weijia Zhang, and Min-Ling Zhang. Multi-instance partial-label learning with margin adjustment. In *Advances in Neural Information Processing Systems 37*, pp. 26331–26354, 2024a.

Wei Tang, Weijia Zhang, and Min-Ling Zhang. Exploiting conjugate label information for multi-instance partial-label learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 4973–4981, 2024b.

Wei Tang, Weijia Zhang, and Min-Ling Zhang. Multi-instance partial-label learning: towards exploiting dual inexact supervision. *Science China Information Sciences*, 67(3):132103, 2024c.

Wei Tang, Yin-Fang Yang, Weijia Zhang, and Min-Ling Zhang. Calibratable disambiguation loss for multi-instance partial-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

Haobo Wang, Mingxuan Xia, Yixuan Li, Yuren Mao, Lei Feng, Gang Chen, and Junbo Zhao. SoLar: Sinkhorn label refinery for imbalanced partial-label learning. In *Advances in Neural Information Processing Systems 35*, pp. 8104–8117, 2022a.

Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, and Junbo Zhao. PiCO: Contrastive label disambiguation for partial label learning. In *International Conference on Learning Representations*, 2022b.

Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. UntrimmedNets for weakly supervised action recognition and detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6402–6411, 2017.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. VideoMAE V2: Scaling video masked autoencoders with dual masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14549–14560, 2023.

Wei Wang, Dong-Dong Wu, Jindong Wang, Gang Niu, Min-Ling Zhang, and Masashi Sugiyama. Realistic evaluation of deep partial-label learning algorithms. In *International Conference on Learning Representations*, 2025.

Yin-Fang Yang, Wei Tang, and Min-Ling Zhang. ProMIPL: A probabilistic generative model for multi-instance partial-label learning. In *2024 IEEE International Conference on Data Mining (ICDM)*, pp. 560–569, 2024.

Yin-Fang Yang, Wei Tang, and Min-Ling Zhang. Fast multi-instance partial-label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 22038–22046, 2025.

Yu Yao, Xizi Wang, Mingze Xu, Zelin Pu, Yuchen Wang, Ella Atkins, and David J. Crandall. DoTA: Unsupervised detection of traffic anomaly in driving videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):444–459, 2023.

Wulian Yun, Mengshi Qi, Chuanming Wang, and Huadong Ma. Weakly-supervised temporal action localization by inferring salient snippet-feature. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 6908–6916, 2024.

Jiahan Zhang, Qi Wei, Feng Liu, and Lei Feng. Candidate Pseudolabel Learning: Enhancing vision-language models by prompt tuning with unlabeled data. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 60004–60020, 2024.

Quan Zhang, Jinwei Fang, Rui Yuan, Xi Tang, Yuxin Qi, Ke Zhang, and Chun Yuan. Weakly supervised temporal action localization via dual-prior collaborative learning guided by multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24139–24148, 2025a.

Quan Zhang, Yuxin Qi, Xi Tang, Rui Yuan, Xi Lin, Ke Zhang, and Chun Yuan. Rethinking pseudo-label guided learning for weakly supervised temporal action localization from the perspective of noise correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 10085–10093, 2025b.

Zhen Zhu, Kai Tang, Songhe Feng, Yixuan Tang, Haobo Wang, Gengyu Lyu, Cheng Peng, and Yining Sun. Dual graph disambiguation for multi-instance partial-label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 29223–29231, 2026.

A Structural Properties and Proofs

We analyze the optimization problems defined in the paper, rather than the convergence of network training. Throughout this appendix, one bag has $n \geq 1$ valid temporal positions, finite costs C_{ckt}, $K \geq 1$ prototypes per class, and a nonempty admissible class set $\mathcal{A}$. We fix $\varepsilon > 0$, $\lambda \geq 0$, and $u_t = 1/n$. Padding positions are excluded from this domain. All plans below are exact optima unless numerical approximations are explicitly discussed. Derivative statements additionally assume that the actor is differentiable at the parameter point under consideration. These properties do not imply recovery of the true label, correct temporal localization, or generalization of the learned network.

A.1 Factorization and Uniqueness

Write $\Pi_{ckt} = m_t \rho_{ckt}$, where $\sum_t m_t = 1$ and $\sum_{c \in \mathcal{A}, k} \rho_{ckt} = 1$. For any positive m_t, conditional minimization over $\rho_{\cdot\cdot t}$ gives

$$Z_t^{\mathcal{A}} = \sum_{c \in \mathcal{A}, k} e^{-C_{ckt}/\varepsilon}, \qquad \rho_{ckt}^{\mathcal{A}} = \frac{e^{-C_{ckt}/\varepsilon}}{Z_t^{\mathcal{A}}}, \qquad f_t^{\mathcal{A}} = -\varepsilon \log Z_t^{\mathcal{A}}. \tag{22}$$

Indeed, its cost plus conditional entropy equals $f_t^{\mathcal{A}} + \varepsilon D_{\mathrm{KL}}(\rho_{\cdot\cdot t} \| \rho_{\cdot\cdot t}^{\mathcal{A}})$. The full objective therefore reduces, up to a constant independent of m, to

$$G_\lambda(m) = \langle f, m \rangle + \varepsilon \sum_t m_t \log(m_t/u_t) + \frac{\lambda}{2} R(m), \qquad R(m) = n \sum_t m_t^2 - 1, \qquad m \in \Delta_n. \tag{23}$$

This is an optimization over an n-dimensional vector subject to one simplex constraint; only its normalization multiplier is a scalar root variable.

Proposition 1 (Unique Interior Optimum)**.** *For finite f, the objective in Eq. (23) has a unique minimizer $m^\star$ with $m_t^\star > 0$ for every valid t. Its necessary and sufficient optimality conditions are*

$$f_t + \varepsilon \log(m_t^\star/u_t) + \lambda n m_t^\star = \xi, \qquad \sum_t m_t^\star = 1. \tag{24}$$

Proof. The continuous extension of $x \log(x/u)$ at zero is strictly convex on $[0, \infty)$. For positive arguments, this follows from its second derivative $1/x$; if one argument is zero, for $y > 0$ and $a \in (0, 1)$, $(ay)\log(ay/u) = ay \log(y/u) + ay \log a < ay \log(y/u)$. Thus, the entropy is strictly convex on the whole simplex. The other terms are linear or convex, and compactness gives existence and uniqueness. A boundary point with $m_t = 0$ can be perturbed by transferring $\delta > 0$ mass from a positive component to t. Its objective difference is $\varepsilon\delta \log \delta + O(\delta) < 0$ for small δ, so it cannot minimize the objective. At the interior optimum, differentiating the Lagrangian and absorbing its common entropy constant into ξ gives Eq. (24). Strict convexity makes these conditions sufficient. For $n = 1$, the unique feasible point is already positive. □

For $\lambda > 0$, the principal Lambert-W function yields

$$m_t(\xi) = \frac{\varepsilon}{\lambda n} W\left(\frac{\lambda}{\varepsilon} e^{(\xi - f_t)/\varepsilon}\right), \qquad H(\xi) = \sum_t m_t(\xi) - 1. \tag{25}$$

Implicit differentiation of Eq. (24) at fixed f gives

$$H'(\xi) = \sum_t \frac{m_t(\xi)}{\varepsilon + \lambda n m_t(\xi)} > 0. \tag{26}$$

The sum in H tends to zero as $\xi \to -\infty$ and to infinity as $\xi \to \infty$, so the root is unique. It obeys the finite bracket

$$\min_t f_t + \lambda \;\leq\; \xi^\star \;\leq\; \max_t f_t + \lambda. \tag{27}$$

To see this, the left side of Eq. (24), as a function of m_t, is strictly increasing and equals $f_t + \lambda$ at $m_t = u_t$. At the lower bracket, each $m_t(\xi) \leq u_t$, and at the upper bracket, each $m_t(\xi) \geq u_t$.

Eq. (27) also provides a valid bracket for a safeguarded Newton–bisection solver. Our implementation instead uses the fixed-iteration Newton procedure described in Appendix B.3, without bisection. If all f_t are equal, the bracket is a single exact root.

The case $\lambda = 0$ must be handled separately:

$$m_t^\star = \frac{e^{-f_t/\varepsilon}}{\sum_s e^{-f_s/\varepsilon}}, \qquad \xi^\star = -\varepsilon \log \sum_t u_t e^{-f_t/\varepsilon}. \tag{28}$$

It retains entropic smoothing. Substitution of zero into the expression with $1/\lambda$ in Eq. (25) is invalid.

A.2 Candidate-Restricted Re-solving and Global Masking

Let Π^A solve the problem over all classes, and let $\widetilde{\Pi}$ solve it over a nonempty candidate set S, using exactly the same $C, \varepsilon, \lambda, n, u$. Define the globally masked actor plan and its retained conditional mass by

$$a_t = \sum_{c \in S, k} \rho^A_{ckt} > 0, \quad Q = \sum_t m_t^A a_t > 0, \quad \Pi^R_{ckt} = \frac{\mathbf{1}[c \in S]\Pi^A_{ckt}}{Q}, \quad m_t^R = \frac{m_t^A a_t}{Q}. \tag{29}$$

All comparisons with $\widetilde{\Pi}$ extend its noncandidate entries by zero. Thus global masking generally changes temporal occupancy. This operation differs both from masking only a class probability vector and from renormalizing the conditional class distribution separately at each time while explicitly retaining m^A.

Proposition 2 (When Re-solving Differs From Masking)**.** *If $\lambda = 0$, then $\widetilde{\Pi} = \Pi^R$. If $\lambda > 0$, then $\widetilde{\Pi} = \Pi^R$ if and only if a_t is constant over valid temporal positions.*

Proof. The restricted conditional distribution is $\widetilde{\rho}_{ckt} = \rho^A_{ckt}/a_t$ for $c \in S$, and $\widetilde{f}_t = f_t^A - \varepsilon \log a_t$. The conditional distribution of Π^R is exactly $\widetilde{\rho}$, so only the occupancies can differ. When $\lambda = 0$, Eq. (28) and Eq. (22) imply $\Pi^A_{ckt} \propto e^{-C_{ckt}/\varepsilon}$ globally; restricting and normalizing this Gibbs distribution is the restricted optimum. For $\lambda > 0$, insertion of m^R into the restricted stationarity expression gives

$$\widetilde{f}_t + \varepsilon \log(m_t^R/u_t) + \lambda n m_t^R = \xi^A - \varepsilon \log Q + \lambda n (m_t^R - m_t^A). \tag{30}$$

If Π^R is optimal, the right side must be constant in t. Since $\lambda n > 0$, $m_t^R - m_t^A$ must be constant; its sum is zero, so $m^R = m^A$. Positivity of m^A and Eq. (29) imply $a_t = Q$ for all t. Conversely, constant a_t gives $m^R = m^A$, making Eq. (30) constant. The unique interior optimum established above then implies $\Pi^R = \widetilde{\Pi}$. □

Consequently, re-solving enforces the occupancy-regularized optimality conditions within the candidate domain. Its distinction from global masking does not follow merely from the fact that temporal occupancy changes. The result assumes the uniform prior and exact objectives stated above; it does not assert equivalence for modified priors, regularizers, or solvers.

Why Retain The Association Between Class And Time? As an illustration, consider two generic class–time distributions,

$$U = \begin{pmatrix} 0.4 & 0.1 \\ 0.1 & 0.4 \end{pmatrix}, \qquad V = \begin{pmatrix} 0.1 & 0.4 \\ 0.4 & 0.1 \end{pmatrix}. \tag{31}$$

Both have class and temporal marginals $(0.5, 0.5)$, yet associate each class with a different time. Their class-marginal KL is zero, while $D_{\mathrm{KL}}(U\|V) = 0.6 \log 4 > 0$. This example concerns distributions, not measured video predictions or an asserted actor–teacher pair. It shows why marginal agreement alone leaves the class–time association unspecified. In our construction, proximity to the proposal selects a joint target among the plans that satisfy both marginals.

A.3 TEACHER PROJECTION AND THE EFFECTIVE SUPERVISION SPACE

Let q be the refined candidate-class marginal and $\widetilde{m}$ the current proposal occupancy. Finite costs and $\varepsilon > 0$ give a positive proposal on $S \times \{1, \dots, K\} \times \{1, \dots, n\}$. The stated initialization and EMA with $\mu \in [0, 1)$ give $q_c > 0$ on S. The KL projection with class marginal q and temporal marginal $\widetilde{m}$ has a unique solution. A strictly positive feasible point exists, for example $q_c \widetilde{m}_t / K$; the entropy boundary argument ensures that the minimizing plan is positive on this domain. Its Lagrange stationarity condition gives

$$\Pi^T_{ckt} = \alpha_c \beta_t \widetilde{\Pi}_{ckt}, \qquad \alpha_c, \beta_t > 0. \tag{32}$$

Although the factors have a reciprocal scaling ambiguity, the plan is unique. With $\widetilde{M}_{ct} = \sum_k \widetilde{\Pi}_{ckt}$, alternating marginal scaling is

$$\alpha_c \leftarrow \frac{q_c}{\sum_t \widetilde{M}_{ct} \beta_t}, \qquad \beta_t \leftarrow \frac{\widetilde{m}_t}{\sum_{c \in S} \widetilde{M}_{ct} \alpha_c}. \tag{33}$$

These updates need only the $|S| \times n$ class–time matrix. They can also be implemented in the log domain. If a target class has exactly zero mass in an alternative initialization, remove that row and apply the argument on the positive target support; extend it by zeros afterwards.

Assume that actor and proposal are constructed at the same parameter point θ_0, using the same costs, prototypes, features, and ε, and that the teacher correction depends only on (c, t). Then

$$r^A_{k|c,t} = r^T_{k|c,t} = \frac{e^{-C_{ckt}/\varepsilon}}{\sum_{k'} e^{-C_{ck't}/\varepsilon}} \quad (c \in S). \tag{34}$$

The temporal multiplier in the actor, the restricted partition function in the proposal, and the teacher scaling in Eq. (32) all cancel when conditioning on (c, t).

Proposition 3 (KL and Current-point Gradient Equivalence)**.** *Set $M^{A/T}_{ct} = \sum_k \Pi^{A/T}_{ckt}$ and extend teacher plans and marginals with zeros outside S (denoted by hats). Under the conditions just stated,*

$$D_{\mathrm{KL}}(\widehat{\Pi}^T \| \Pi^A) = D_{\mathrm{KL}}(\widehat{M}^T \| M^A), \tag{35}$$

$$= D_{\mathrm{KL}}(q^T \| q^A) + \sum_{c : q^T_c > 0} q^T_c D_{\mathrm{KL}}\big(P_T(t \mid c) \| P_A(t \mid c)\big), \tag{36}$$

where $q^T_c = \sum_t \widehat{M}^T_{ct}$ is the actual teacher class marginal. If this teacher is frozen, the three-dimensional and two-dimensional losses also have equal first derivatives with respect to actor parameters at $\theta = \theta_0$.

Proof. The KL chain rule applied to (c, t) and then to k gives

$$D_{\mathrm{KL}}(\widehat{\Pi}^T \| \Pi^A) = D_{\mathrm{KL}}(\widehat{M}^T \| M^A) + \sum_{c,t : \widehat{M}^T_{ct} > 0} \widehat{M}^T_{ct} D_{\mathrm{KL}}(r^T_{\cdot|c,t} \| r^A_{\cdot|c,t}). \tag{37}$$

Eq. (34) makes the last term zero. Applying the same chain rule to class and time in M proves Eq. (36). For the derivative statement, denote the teacher built at θ_0 by T_0 and hold it fixed while differentiating the actor. The difference between the two loss gradients is

$$-\sum_{c,t} \widehat{M}^{T_0}_{ct} \sum_k r^{T_0}_{k|c,t} \nabla_\theta \log r^A_{k|c,t}(\theta). \tag{38}$$

At θ_0, $r^{T_0} = r^A(\theta_0)$, so every inner sum is $\sum_k \nabla_\theta r^A_{k|c,t}(\theta_0) = \nabla_\theta 1 = 0$. This proves local first-order equivalence despite stopping gradients through the teacher; equality of values alone would not justify that derivative conclusion. □

The extra supervision beyond the class marginal is therefore the class-conditional temporal distribution, parameterized by multi-prototype geometry. Prototype parameters still influence $\sum_k e^{-C_{ckt}/\varepsilon}$ and generally receive nonzero gradients. There is no additional nonzero prototype-conditional KL term under these conditions. The result does not extend to stale teachers reused after actor updates, different teacher costs or temperatures, prototype-dependent corrections, or higher derivatives of a frozen-teacher loss. For an approximate multiplicative projection, conditional preservation still holds, but Eq. (36) uses its actual marginal q^T, which need not equal the requested q until projection residuals vanish.

A.4 Teacher Projection and Class–Time Supervision

Proposition 4 (No Momentum And No Occupancy Penalty)**.** *Under the same-cost and exact-solution assumptions, let $\mu = 0$ and $\lambda = 0$. Then the teacher equals the candidate proposal and*

$$D_{\mathrm{KL}}(\widehat{\Pi}^T \| \Pi^A) = -\log Q, \qquad Q = \sum_{c \in S} q_c^A. \tag{39}$$

The stopped-teacher gradient at its construction point also equals the gradient of $-\log Q$.

Proof. When $\mu = 0$, the refined marginal is the proposal class marginal. The proposal already satisfies both teacher constraints and has zero KL distance to itself, so the uniqueness of the projection implies $\Pi^T = \widetilde{\Pi}$. The masking proposition with $\lambda = 0$ gives $\widehat{\Pi}^T = \mathbf{1}[c \in S]\Pi^A/Q$. The log ratio on its support is the constant $-\log Q$, proving the value identity. At the teacher construction point, differentiation with that teacher fixed gives

$$-\sum_{c \in S,k,t} \frac{\Pi^A_{ckt}}{Q} \nabla_\theta \log \Pi^A_{ckt} = -\frac{1}{Q} \nabla_\theta \sum_{c \in S,k,t} \Pi^A_{ckt} = \nabla_\theta[-\log Q]. \tag{40}$$

No derivative through the teacher is used. □

This is a boundary configuration, not an identity for the full method with positive momentum or occupancy regularization. It motivates a controlled candidate-mass baseline without asserting its relative empirical performance.

A.5 Effective-Support Control by Occupancy Regularization

Define $N_{\mathrm{eff}}(m) = 1/\sum_t m_t^2$, so that $1 \le N_{\mathrm{eff}}(m) \le n$ and

$$R(m) = \frac{n}{N_{\mathrm{eff}}(m)} - 1 = \chi^2(m\|u) = \exp(D_2(m\|u)) - 1. \tag{41}$$

Proposition 5 (Fixed-geometry Monotonicity)**.** *For fixed $f, \varepsilon > 0, n$, $R(m^\star_\lambda)$ is nonincreasing and $N_{\mathrm{eff}}(m^\star_\lambda)$ is nondecreasing in $\lambda \ge 0$. Moreover, $m^\star_\lambda \to u$ as $\lambda \to \infty$.*

Proof. Write $F(m) = \langle f, m\rangle + \varepsilon D_{\mathrm{KL}}(m\|u)$. For $\lambda_2 > \lambda_1 \ge 0$, optimality at the two coefficients gives

$$F(m_1) + \tfrac{\lambda_1}{2} R(m_1) \le F(m_2) + \tfrac{\lambda_1}{2} R(m_2),$$
$$F(m_2) + \tfrac{\lambda_2}{2} R(m_2) \le F(m_1) + \tfrac{\lambda_2}{2} R(m_1).$$

Adding them yields $(\lambda_2 - \lambda_1)(R(m_2) - R(m_1)) \le 0$. Eq. (41) gives the support statement. Since $R(u) = 0$, comparison with the feasible u gives, for $\lambda > 0$,

$$0 \le R(m^\star_\lambda) \le \frac{2[F(u) - \min_{m \in \Delta_n} F(m)]}{\lambda}. \tag{42}$$

The numerator is finite by compactness. As $R(m) = n\|m - u\|_2^2$, the bound implies convergence to u. At $\lambda = 0$, Eq. (28) gives the entropy-smoothed Gibbs occupancy. □

This statement holds with geometry fixed. During learning f changes, so it does not establish monotonic support across training updates or monotonic accuracy in λ. Effective support measures mass concentration, not semantic diversity or true-action coverage; overly diffuse mass can also assign weight to the background around a short action.

A.6 Differentiability and Computational Complexity

For an exact interior solution, set $w_t = (\varepsilon/m_t^\star + \lambda n)^{-1} > 0$. Differentiating Eq. (24) gives $dm_t = w_t(d\xi - df_t)$, and $\sum_t dm_t = 0$ implies $d\xi = (\sum_t w_t df_t)/(\sum_t w_t)$. Hence

$$\frac{\partial m^\star}{\partial f} = -\operatorname{diag}(w) + \frac{ww^\top}{\sum_t w_t}. \tag{43}$$

This matrix is symmetric and negative semidefinite: for any vector v, its quadratic form is the negative weighted variance $-\sum_t w_t v_t^2 + (\sum_t w_t v_t)^2/\sum_t w_t \le 0$. Its null direction $\mathbf{1}$ reflects invariance to adding a common constant to f. The formula also covers $\lambda = 0$, yielding the usual softmax Jacobian with a negative inverse-temperature factor. A Jacobian–vector product costs $O(n)$ without materializing an $n \times n$ matrix. Differentiation of the complete actor must additionally include $\partial f_t/\partial C_{ckt} = \rho_{ckt}$ and the conditional Gibbs derivative; retaining only the occupancy path would change the gradient.

For feature dimension d, the dense spherical matching costs $O(CKnd)$ operations and $O(CKn)$ storage when scores are materialized. Given these scores, the conditional distributions and free energies cost $O(CKn)$. With J_A and J_S scalar root iterations, the actor and proposal occupancy solves cost $O((J_A + J_S)n)$, in addition to the $O(|S|Kn)$ restricted reductions. With I marginal-scaling iterations, projection on $\widetilde{M}$ costs $O(I|S|n)$; a three-dimensional teacher, if materialized, costs $O(|S|Kn)$ storage. These counts exclude the temporal encoder and depend on the requested numerical precision. The scalar normalization root does not remove the class–prototype matching cost.

For numerical evaluation, define $h_t = \log(\lambda/\varepsilon) + (\xi - f_t)/\varepsilon$ and let $\omega_t = W(e^{h_t})$. Rather than explicitly forming e^{h_t}, we evaluate ω_t from $\omega_t + \log\omega_t = h_t$ using stable Newton updates and recover $m_t = \varepsilon\omega_t/(\lambda n)$. The implementation uses eight fixed Lambert-W updates per evaluation and twelve Newton iterations for the normalization root, with the latter performed in double precision. The $\lambda = 0$ branch uses the closed-form softmax solution.

An approximate solver should report its mass residual $|\sum_t m_t - 1|$, stationarity residual $\max_t |f_t + \varepsilon\log(m_t/u_t) + \lambda n m_t - \xi|$, and both teacher marginal residuals $\|\sum_{k,t}\Pi^T_{ckt} - q_c\|_\infty$ and $\|\sum_{c,k}\Pi^T_{ckt} - \widetilde{m}_t\|_\infty$.

B Dataset construction and implementation details

B.1 More Information on Benchmark Construction

We construct VCMIPL benchmarks from Breakfast (Kuehne et al., 2014), DoTA (Yao et al., 2023), and FineAction (Liu et al., 2022). Table 3 summarizes the video counts, class counts, and candidate-set sizes. Each example consists of an ordered video bag and a candidate label set containing its ground-truth class. For FineAction, we retain videos with a single annotated target class and exclude classes containing fewer than 50 videos. The resulting benchmarks contain 1,989 videos from 10 classes for Breakfast, 4,577 videos from 9 classes for DoTA, and 10,675 videos from 78 classes for FineAction. In the output space, we generate candidate labels using a VideoMAEv2 recognizer equipped with a TimeSformerHead in MMAction2 (MMAction2 Contributors, 2020). Candidate probabilities are obtained through five-fold cross-fitting: the recognizer is fine-tuned using the exact video labels from four folds and predicts the remaining fold. Repeating this procedure produces an out-of-fold prediction for every video. The recognizer is fine-tuned for 30 epochs on DoTA and 100 epochs on both Breakfast and FineAction.

Following the sample-wise confidence accumulation and class-wise selection principles of CPL (Zhang et al., 2024), we combine two criteria. Let $\mathbf{p}_i = (p_{i1}, \dots, p_{iC})$ denote the out-of-fold class-probability vector for video i, and let $c_{i1}, \dots, c_{iC}$ order the classes by decreasing confidence. The sample-wise candidate set contains the smallest number of classes whose cumulative probability reaches 95%:

$$A_i = \{c_{i1}, \dots, c_{i\ell_i}\}, \qquad \ell_i = \min\left\{\ell \in \{1, \dots, C\} : \sum_{j=1}^{\ell} p_{i,c_{ij}} \ge 0.95\right\}. \tag{44}$$

Table 3: Video-level statistics of the constructed MIPL benchmarks.

Dataset	Videos	Classes	Mean $\lvert S_i \rvert$	Range $\lvert S_i \rvert$
Breakfast	1,989	10	3.3896	1–9
DoTA	4,577	9	3.2733	1–8
FineAction	10,675	78	11.5592	2–64

For each class c, let B_c denote the indices of the top 50% of videos in the class-wise ranking population, ordered by their confidence for that class. Intersecting the two selections gives

$$\widetilde{S}_i = A_i \cap \{c \in \mathcal{Y} : i \in B_c\}. \tag{45}$$

To satisfy the standard MIPL assumption, we include the ground-truth class whenever it is absent:

$$S_i = \widetilde{S}_i \cup \{Y_i\}. \tag{46}$$

Consequently, the final candidate set is nonempty and satisfies $Y_i \in S_i$ by construction. This guarantee applies to the corrected candidate sets, rather than to the recognizer's predictions before correction. Exact labels are used to train the candidate recognizer and perform this inclusion step during benchmark construction; downstream MIPL learning uses only the resulting candidate sets as supervision.

B.2 Feature Representations and Dataset Statistics

ResNet (He et al., 2016) and DINOv3 (Siméoni et al., 2025) provide frame-level features, whereas SlowFast (Feichtenhofer et al., 2019) and VideoMAEv2 (Wang et al., 2023) provide clip-level features. These representations yield twelve dataset–feature configurations spanning different temporal granularities and feature dimensions.

Table 4 reports the complete instance statistics. Total instances denotes the number of extracted frame or clip features across all videos; maximum, minimum, and mean refer to the number of instances per video. Dimension denotes the width of each feature vector. Within each dataset, the four representations share the video, class, and candidate-set statistics in Table 3. The fine-tuning budgets in the dataset headers refer to the candidate recognizer, not to the individual feature extractors.

Table 4: Instance and feature statistics for all twelve dataset–feature configurations.

Feature	Total instances	Maximum	Minimum	Mean	Dimension
Breakfast: 1,989 videos, 10 classes, 100 fine-tuning epochs Candidate size: 1–9, mean 3.3896					
ResNet	4,103,316	9,745	186	2,063.00	2,048
DINOv3	4,103,316	9,745	186	2,063.00	4,096
SlowFast	1,026,577	2,437	47	516.13	2,304
VideoMAEv2	1,026,577	2,437	47	516.13	768
DoTA: 4,577 videos, 9 classes, 30 fine-tuning epochs Candidate size: 1–8, mean 3.2733					
ResNet	469,756	300	26	102.63	2,048
DINOv3	469,756	300	26	102.63	4,096
SlowFast	119,188	75	7	26.04	2,304
VideoMAEv2	119,188	75	7	26.04	768
FineAction: 10,675 videos, 78 classes, 100 fine-tuning epochs Candidate size: 2–64, mean 11.5592					
ResNet	37,723,340	64,644	36	3,533.80	2,048
DINOv3	37,723,340	64,644	36	3,533.80	4,096
SlowFast	9,518,494	47,372	9	891.66	2,304
VideoMAEv2	9,518,494	47,372	9	891.66	768

Table 5: Experimental settings and configurable implementation defaults of PIVOTMIPL.

Setting	Value or implementation
Framework / hardware	PyTorch / NVIDIA H100
Optimizer	AdamW; $(\beta_1, \beta_2) = (0.9, 0.999)$; weight decay 0
Learning rates	Prototypes: 5×10^{-3}; temporal encoder: 5×10^{-4}
Gradient clipping	Global ℓ_2 norm clipped to 5.0
Training budget	40 epochs
Motion module	Average-filter kernel 3; convolution kernel 5; dilations $1, 2$; motion-residual dropout 0.1
Candidate recognizer budget	DoTA: 30 epochs; Breakfast and FineAction: 100 epochs
Main evaluation	Five folds; mean and standard deviation of ACC and BACC
Ablation evaluation	Fold 1 ACC at epoch 40
Learning-rate schedule	Constant; no warm-up or scheduler
Batch size	One complete video bag per optimization step
Prototypes per class	Breakfast: $K = 4$; DoTA: $K = 8$; FineAction: $K = 12$
Class momentum	$\mu = 0.9$
Transport regularization	Entropic coefficient $\varepsilon = 0.07$; occupancy coefficient $\lambda = 0.01$
Temporal encoder	Hidden dimension 512; 4 bidirectional Mamba-2 blocks
Feed-forward network	Expansion factor 4; intermediate dimension 2048
Mamba-2 configuration	State dimension 128; convolution width 4; expansion factor 2
Mamba-2 computation	Head dimension 64; chunk size 256; memory-efficient path enabled
Teacher projection	At most 120 IPFP iterations; marginal tolerance 10^{-6}
Base random seed	123, with fold-dependent offsets

B.3 MORE IMPLEMENTATION SETTINGS

We adopt a fixed five-fold evaluation protocol for all three benchmarks. In each fold, four predefined folds are used for training and the remaining fold for testing, and the reported ACC and BACC are averaged over the five held-out folds. Candidate labels are generated in an out-of-fold manner using the same fold partition: the candidate recognizer is fine-tuned on four folds and applied to the held-out fold, and this procedure is repeated until every video receives an out-of-fold prediction. The downstream MIPL models are then trained using only the resulting candidate label sets as supervision. PIVOTMIPL is optimized for 40 epochs with one complete video bag per optimization step, without temporal padding, and the same training protocol is used across all feature representations unless otherwise specified. Ablation studies are conducted on Fold 1 and evaluated at epoch 40. Detailed optimization, architecture, transport, and teacher-projection settings are summarized in Table 5.

C ADDITIONAL EXPERIMENTAL RESULTS

C.1 BALANCED ACCURACY RESULTS

Table 6 further confirms the advantage of PIVOTMIPL under class-balanced evaluation. PIVOTMIPL achieves the highest mean BACC, the mean recall over ground-truth classes represented in each test fold, in 10 of the 12 dataset–feature configurations, with an average gain of 3.26% over the strongest baseline. Notably, the improvement pattern closely follows that of ACC, indicating that the gains are not primarily driven by better prediction on dominant classes, but extend across the class distribution. PIVOTMIPL also ranks first in all four DoTA-MIPL configurations, where the substantially lower BACC values make balanced recognition particularly challenging. The only exceptions again occur with VideoMAEv2, suggesting that the reduced headroom observed in the ACC results persists under class-balanced evaluation.

C.2 EFFICIENCY ANALYSIS

We also evaluate the efficiency of PIVOTMIPL, presented in Figure 2. Figure 2 compares classification accuracy against training time. PIVOTMIPL exhibits a favorable accuracy–efficiency trade-off

Table 6: Balanced accuracy on the benchmark datasets. Results are reported as (mean ± std %). The best result in each column is bolded, and the second-best results are underlined.

Dataset	Method	DINOv3	VideoMAEv2	ResNet	SlowFast
Breakfast -MIPL	DEMIPL	10.2% ± 0.4%	24.4% ± 10.2%	10.6% ± 0.6%	10.0% ± 0.0%
	ELIMIPL	27.9% ± 0.8%	52.4% ± 2.1%	26.0% ± 1.9%	10.0% ± 0.0%
	MIPLMA	34.2% ± 1.5%	63.8% ± 2.0%	30.8% ± 2.6%	10.0% ± 0.0%
	PROMIPL	<u>61.9%</u> ± 1.1%	83.3% ± 3.5%	<u>43.7%</u> ± 1.0%	41.4% ± 1.1%
	PSMIPL	47.7% ± 4.9%	70.1% ± 7.1%	10.9% ± 1.4%	32.9% ± 0.5%
	DUALG	45.2% ± 2.7%	65.1% ± 4.1%	34.7% ± 2.2%	13.8% ± 1.7%
	AGOPMIPL	46.0% ± 4.3%	**87.2%** ± 4.3%	33.3% ± 3.3%	<u>50.1%</u> ± 2.4%
	PIVOTMIPL	**66.0%** ± 1.0%	<u>85.2%</u> ± 4.3%	**52.2%** ± 3.3%	**55.8%** ± 1.8%
DoTA -MIPL	DEMIPL	24.8% ± 0.7%	42.7% ± 0.5%	23.2% ± 0.4%	18.2% ± 2.6%
	ELIMIPL	24.8% ± 0.8%	42.9% ± 0.8%	23.9% ± 0.6%	21.2% ± 1.7%
	MIPLMA	25.8% ± 0.6%	43.2% ± 0.5%	23.9% ± 0.5%	21.5% ± 3.4%
	PROMIPL	<u>32.7%</u> ± 1.6%	58.0% ± 1.5%	<u>28.8%</u> ± 2.0%	<u>28.8%</u> ± 0.8%
	PSMIPL	30.0% ± 0.9%	<u>58.3%</u> ± 1.4%	26.2% ± 0.5%	14.9% ± 4.7%
	DUALG	26.0% ± 0.7%	44.1% ± 0.6%	24.9% ± 0.8%	24.2% ± 0.9%
	AGOPMIPL	28.6% ± 0.9%	51.4% ± 3.1%	26.1% ± 1.0%	27.5% ± 1.0%
	PIVOTMIPL	**33.8%** ± 1.5%	**63.2%** ± 3.5%	**31.8%** ± 1.2%	**31.8%** ± 0.7%
FineAction -MIPL	DEMIPL	36.0% ± 1.8%	48.8% ± 2.8%	18.3% ± 1.3%	44.6% ± 2.1%
	ELIMIPL	34.8% ± 1.6%	44.0% ± 2.7%	18.9% ± 2.0%	44.7% ± 1.4%
	MIPLMA	49.6% ± 2.0%	61.1% ± 0.3%	30.6% ± 2.7%	51.4% ± 2.2%
	PROMIPL	79.0% ± 1.3%	91.0% ± 3.5%	66.5% ± 1.6%	80.2% ± 2.0%
	PSMIPL	57.2% ± 1.1%	63.1% ± 1.6%	39.4% ± 1.8%	29.2% ± 14.1%
	DUALG	53.7% ± 0.6%	66.3% ± 1.2%	40.3% ± 1.1%	57.4% ± 0.7%
	AGOPMIPL	<u>81.7%</u> ± 1.1%	**95.2%** ± 1.2%	<u>67.5%</u> ± 1.6%	<u>82.0%</u> ± 3.0%
	PIVOTMIPL	**86.3%** ± 1.5%	<u>92.0%</u> ± 1.7%	**73.6%** ± 1.1%	**85.3%** ± 1.4%

across most dataset–feature configurations; the exceptions occur in the VideoMAEv2 settings where AGOPMIPL can achieve a Pareto-superior operating point.

C.3 Ablation Analysis

C.3.1 Ablation Summaries

Table 7 shows that the gains of PIVOTMIPL are distributed across multiple components rather than dominated by a single design choice. Replacing the plan-level objective with class-level KL (A3) or removing the temporal-marginal constraint from the teacher (A4) degrades performance in all 12 configurations, highlighting joint class–time supervision and dual-marginal projection as the most structurally consistent contributors. Removing the temporal encoder (A6) causes the largest average drop, while the strong performance gap retained against encoder-matched baselines indicates that temporal contextualization alone does not explain the full advantage. Finally, A1, A2, and A5 show that occupancy regularization, adaptive temporal allocation, and multiple prototypes each provide complementary gains, supporting the need to model both temporal concentration and intra-class variation.

Table 7: Summary of the ablations across twelve dataset–feature configurations.

Variant	Modification	Full-model wins	Mean gain
A1	No occupancy regularization	10/12	+3.25%
A2	Uniform temporal occupancy	9/12	+0.72%
A3	Class-level KL	12/12	+3.26%
A4	Class-marginal-only teacher	12/12	+2.53%
A5	Single prototype per class	11/12	+2.52%
A6	No temporal encoder	11/12	+5.77%

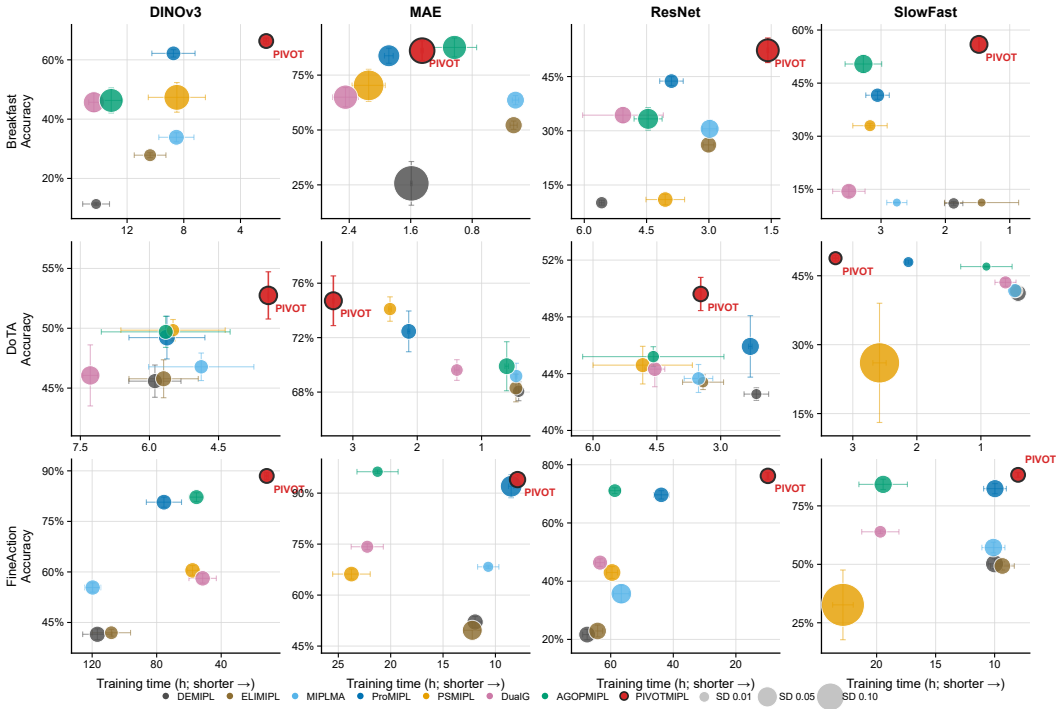

Figure 2: Accuracy–training-time trade-off across the twelve dataset–feature configurations. Each panel corresponds to one dataset–feature setting and compares PIVOTMIPL with the MIPL baselines in terms of mean classification accuracy and total training time over five folds. Higher accuracy and shorter training time are preferred.

Table 8: Accuracy for AGOPMIPL, AGOPMIPL augmented with the same temporal encoder architecture as PIVOTMIPL, and PIVOTMIPL. The best and second-best values within each row are shown in bold and underlined, respectively.

Dataset	Feature	AGOPMIPL	AGOPMIPL + Temporal Encoder	PIVOTMIPL
Breakfast	DINOv3	46.4%	<u>53.1%</u>	**66.4%**
	VideoMAEv2	**79.3%**	<u>78.1%</u>	77.4%
	ResNet	31.4%	<u>45.4%</u>	**52.2%**
	SlowFast	50.6%	<u>53.1%</u>	**53.9%**
DoTA	DINOv3	<u>49.2%</u>	48.4%	**54.3%**
	VideoMAEv2	69.5%	<u>69.9%</u>	**72.8%**
	ResNet	46.0%	<u>48.6%</u>	**50.9%**
	SlowFast	46.9%	<u>47.6%</u>	**48.4%**
FineAction	DINOv3	85.5%	<u>86.8%</u>	**90.0%**
	VideoMAEv2	<u>96.7%</u>	**97.2%**	92.4%
	ResNet	73.1%	<u>75.4%</u>	**78.1%**
	SlowFast	87.9%	<u>89.2%</u>	**90.0%**

C.3.2 ANALYSIS OF THE EFFECTIVENESS OF THE TEMPORAL ENCODER

To examine whether adding temporal encoding alone can reproduce the performance advantage of PIVOTMIPL, we equip AGOPMIPL, the strongest baseline, with the same temporal encoder architecture. We compare the original AGOPMIPL, its encoder-augmented variant, and PIVOTMIPL. The result of the above shows PIVOTMIPL retains higher ACC in 10 configurations, with an average advantage of 2.83% over the encoder-augmented baseline. Together with A6, this comparison suggests that temporal encoding contributes to performance, but adding the same encoder architecture to AGOPMIPL does not by itself account for the observed advantage of PIVOTMIPL across most configurations.

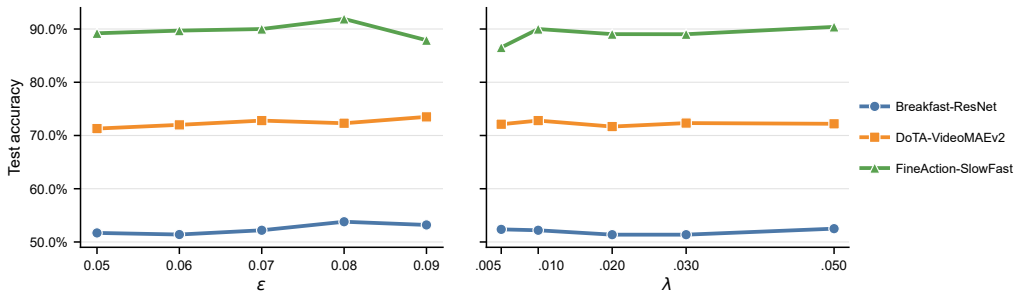

Figure 3: Sensitivity of PIVOTMIPL to the entropic coefficient ε and occupancy coefficient λ on three representative dataset–feature configurations.

C.4 PARAMETER SENSITIVITY ANALYSIS

We examine the sensitivity of PIVOTMIPL to the two transport-specific hyperparameters, the entropic coefficient ε and the occupancy coefficient λ in Eq. (8), using three representative Fold-1 configurations: Breakfast–ResNet, DoTA–VideoMAEv2, and FineAction–SlowFast. As shown in Figure 3, performance varies only moderately over the evaluated ranges for both parameters, despite substantial differences in dataset scale and feature representation. The two parameters play complementary roles in the transport inference: ε controls the sharpness of the class–prototype–time assignment, whereas λ regulates the concentration of the temporal marginal. A very small ε favors sharper matching, while increasing it produces smoother assignments; similarly, increasing λ discourages excessive concentration of mass on a small number of temporal positions. The relatively broad performance plateaus therefore suggest that PIVOTMIPL does not depend on a narrowly tuned balance between these two effects. Instead, the structured teacher and plan-level supervision remain effective under a range of assignment smoothness and temporal occupancy regularization strengths. This robustness is particularly desirable in VCMIPL, where sequence length, temporal redundancy, and feature granularity vary substantially across datasets.

D LIMITATIONS

Our study focuses on offline VCMIPL within a fixed label space. Although the experiments span multiple datasets and feature representations, they do not establish robustness to distribution shifts or evolving class vocabularies. Assessing how class–time associations transfer across such changes remains a direction for future work. Moreover, the current formulation uses bidirectional temporal context and normalizes assignments over the observed video. Extending it to strictly online prediction would require causal contextualization and incremental assignment updates that account for incomplete temporal evidence.