Which Landmarks does Isolated Sign Language Recognition Actually Need?

Anonymous ACL submission

Abstract

Sign languages are fully natural human languages articulated through hands, face, and body in three-dimensional space, yet isolated sign language recognition (ISLR) systems routinely feed all available pose landmarks to their encoders without asking which are actually needed. We introduce a learnable L_0-regularized landmark-pruning framework that attaches a per-landmark gate to any skeleton encoder and traces a continuous Pareto frontier between landmark count and recognition accuracy end-to-end, without manual graph design or hand-picked landmark subsets. Across three sign languages we find that 24–48 of the 543 MediaPipe landmarks suffice to match or exceed the full-skeleton baseline, and that as few as 10 landmarks remain competitive on three of the four settings we evaluate. The selected subsets are substantially shared across languages despite their independent origins; encoders spanning the dominant skeleton families (GRU, ST-GCN, SPOTER) benefit unequally from pruning but converge to similar accuracy. We interpret this convergence as evidence that the informative skeletal subset is a property of sign language as a linguistic system rather than of any particular language.

1 Introduction

Natural language processing has largely been shaped around written language: discrete tokens, sequential orderings, and the vast textual corpora scraped from the web. Sign languages fit none of these assumptions. They are fully natural human languages (Stokoe, 2005) articulated simultaneously through the hands, face, and body in three-dimensional space (Battison, 1977), with no widely adopted written form and no clear word boundaries in the continuous visual signal. Rather than traditional phonemes, sign languages use four phonological parameters (handshape, location, movement, orientation) as well as non-manual markers (Sandler, 2012). Building language technology that serves signers therefore requires representations that treat the body itself as the input, with direct accessibility stakes for Deaf and hard-of-hearing users.

There is no single "sign language." Hundreds of distinct sign languages are used around the world. The three we evaluate—American Sign Language (ASL), Turkish Sign Language (TID), and Greek Sign Language (GSL)—emerged independently of the spoken languages around them and of each other, and are mutually unintelligible. Most Isolated Sign Language Recognition (ISLR) work consequently trains and evaluates within a single language, since the corpora for each language are gathered independently and do not share vocabularies.

Isolated Sign Language Recognition (ISLR), the task of classifying a short video clip into a fixed gloss vocabulary, has used either RGB approaches that operate on raw video (De Coster et al., 2021) or skeleton-based approaches that operate on pose-extractor landmarks (Li et al., 2020). Skeleton input is attractive because it abstracts away from signer appearance, is lighter to train, and admits a range of encoders (recurrent, graph-convolutional, Transformer-based). Throughout this paper we focus on skeleton-based approaches and ask a basic question: which landmarks does the model actually need?

We formulate landmark selection as a learnable L_0-regularized gating problem over the full 543-landmark MediaPipe Holistic output (Lugaresi et al., 2019), tracing a continuous Pareto frontier between landmark count and recognition accuracy end-to-end. Across ISLR datasets for three typologically unrelated sign languages (AUTSL, ASL Citizen, and the VCL GSL corpus) and four skeleton encoder families (BERT, GRU, ST-GCN, and SPOTER), we find that 24–48 of the 543 landmarks suffice to match or exceed the full-skeleton baseline, that as few as 10 landmarks remain competitive on three of the four datasets, and that the se-

lected subset is largely shared across languages. We find this convergence to be evidence that the informative landmark subset is a property of sign language as a linguistic system rather than of any particular sign language.

Contributions. We contribute (i) a learnable L_0 landmark-pruning framework that traces the landmark-count/accuracy Pareto frontier end-to-end, without manual graph design or hand-picked landmark subsets; (ii) empirical evidence across three sign languages that 24–48 of the 543 MediaPipe landmarks suffice for isolated sign recognition; and (iii) a cross-lingual landmark-overlap analysis grounding a linguistic interpretation of which landmarks matter.

2 Related Work

2.1 Skeleton-Based ISLR

Skeleton-based isolated sign language recognition operates on the sequence of body, hand, and face landmarks extracted from each frame, sidestepping the appearance and signer-identity variation that complicates pixel-level approaches. The standard benchmark for English-language ISLR is WLASL, introduced alongside Pose-TGCN as a baseline (Li et al., 2020). Subsequent work has explored a range of encoders over the resulting landmark sequences. Spatial-temporal graph convolutional networks (ST-GCN) (Yan et al., 2018), originally proposed for general action recognition, have been adapted extensively to sign language by introducing anatomically motivated adjacencies. SAM-SLR (Jiang et al., 2021) combines a multi-stream skeleton GCN with sparse-spatial-temporal convolutions. Notably, the authors attribute the largest portion of their accuracy gain to reducing HRNet's (Wang et al., 2020) 133 raw whole-body keypoints to a 27-node skeleton graph chosen by hand. This observation directly motivates the question we ask in this paper.

2.2 Keypoint Selection and Pruning

Pose estimators such as MediaPipe Holistic emit 543 landmarks per frame (468 face, 33 body, 21 per hand). Reducing the number of landmarks fed to a sign-language recognizer is not a new idea, but every prior approach we are aware of selects the subset by hand. SAM-SLR's manual reduction from 133 to 27 keypoints (Jiang et al., 2021) is the most-cited example. The Multi-Stream Keypoint Attention (MSKA) network (Guan et al., 2024) takes a related but more aggressive step, splitting the 133 keypoints into hand, face, and body streams that are processed independently and fused. Preprocessing-focused work for sign-language translation (Kim and Baek, 2023) drops leg landmarks and "noisy" facial points by hand, and finds that each preprocessing step contributes meaningfully to downstream accuracy. These reductions all bake in the designer's prior about which body parts matter for sign, and to our knowledge no prior work learns the subset end-to-end. We instead learn the subset directly from the recognition objective. Our gating mechanism is a sigmoid-based variant of the differentiable L_0 norm proposed by Louizos et al. (2018) for sparse network weights; we apply it to landmark-level inputs rather than to weights.

2.3 Cross-Lingual SLR

A small but growing body of work has examined whether models trained on one sign language transfer to another. OpenHands (Selvaraj et al., 2021) released pose-based pretrained models for ASL, Argentinian, Chinese, Greek, Indian, and Turkish sign languages and demonstrated cross-lingual transfer from Indian SL to other targets. MLSLT (Yin et al., 2022) formulates multilingual sign language translation as a single sequence-to-sequence problem and shows that multilingual training is superior to monolingual for low-resource languages. Wei and Chen (2023) focus more narrowly on visually similar signs across languages and use cross-lingual augmentations to improve continuous SLR. Across these approaches, the gains from multilingual data are evident, but the underlying mechanism is rarely examined. The authors do not address whether transfer arises from shared linguistic structure or from low-level visual features. Our cross-language landmark-overlap analysis offers one concrete probe of that question.

2.4 Linguistic Foundations

Signed languages use four phonological parameters (handshape, location, movement, and orientation) along with non-manual markers that carry grammatical information (Stokoe, 2005; Battison, 1977; Sandler, 2012). Unlike spoken phonology, these elements are produced simultaneously on the hands, body, and face rather than serially. Zeshan (2004) further establishes typological convergence across thirty-five sign languages despite their independent origins. Together this body of work provides the linguistic basis for our hypothesis that the landmark

subset supporting recognition is largely determined by the structure of sign language as a system.

3 Method

3.1 Problem Formulation

Given a video clip of a single sign, the goal of ISLR is to assign one of C class labels (glosses). We assume the video has been processed with MediaPipe Holistic (Lugaresi et al., 2019), producing a sequence of T frames each carrying $J = 543$ landmarks in R^P (where R is the set of real numbers and $P = 2$ for (x, y)). The full input is then a tensor $X \in R^{T \times J \times P}$, where the $J = 543$ landmarks decompose anatomically as 468 face, 33 body, 21 left-hand, and 21 right-hand landmarks. We use *landmark* throughout this paper; in the literature, *keypoint*, *pose point*, and *joint* appear interchangeably for the same objects.

Each input tensor X is gap-filled along the time axis, standardized to zero mean and unit variance using statistics computed over the full 543-landmark skeleton (so the normalization is invariant to which landmarks the gate later selects), restricted to the selected landmark indices, and uniformly subsampled to $T = 16$ frames.

3.2 Learnable Landmark Pruning

The core of our method is a per-landmark learnable gate. We attach a real-valued logit $\alpha_j \in R$ to each of the $J = 543$ landmarks, initialized so that all gates are nearly open at the start of training:

$$\alpha_j^{(0)} = \log\left(\frac{p_0}{1 - p_0}\right), \qquad j = 1, \ldots, J, \tag{1}$$

with $p_0 = 0.98$. Initializing close to fully open avoids gradient starvation: gates that begin near zero receive almost no learning signal and cannot recover, while gates that begin open can be driven shut by the sparsity penalty if they prove uninformative.

The gate. During training, each landmark $x_{t,j} \in R^P$ is multiplied by a soft mask

$$m_j(\tau) \;=\; \sigma\left(\frac{\alpha_j}{\tau}\right), \tag{2}$$

where σ is the logistic sigmoid and $\tau > 0$ is a temperature controlling decision sharpness. The gated input is $\tilde{x}_{t,j} = m_j(\tau) \cdot x_{t,j}$. We anneal τ from $\tau_0 = 10$ at step 0 to $\tau_{\text{end}} = 0.01$ at step $T_\tau = 50{,}000$. Early in training the gate is nearly linear and all landmarks contribute, allowing the encoder to fit the data on the (almost) full skeleton; later in training the gate's response sharpens around the underlying logit α_j, amplifying the differences between informative and uninformative landmarks.

Sparsity penalty. Sparsity is encouraged through a differentiable surrogate of the L_0 norm in the spirit of Louizos et al. (2018). We use a simpler Bernoulli relaxation in which each $\sigma(\alpha_j)$ is treated directly as a per-landmark keep probability, rather than the hard-concrete distribution proposed by Louizos et al. for sparse network weights. The penalty is the expected mass under this relaxation, normalized by both batch size and landmark count:

$$\mathcal{L}_0(\alpha) \;=\; \frac{1}{B \cdot J} \sum_{j=1}^{J} \sigma(\alpha_j). \tag{3}$$

Two design choices matter here. First, the penalty is computed at $\tau = 1$ rather than at the current annealing temperature; this prevents the temperature schedule from interacting with the sparsity pressure and lets the two schedules be tuned independently. Second, dividing by $B \cdot J$ is a numerical-scale choice rather than a derivation. The penalty is per-landmark, not per-example, so the $1/B$ factor is not implied by the surrogate itself; we include it to keep the penalty on the same numerical scale as the per-example cross-entropy loss. Without it, the raw sum (which is on the order of J) overwhelms the classification signal and the model collapses to all-zero gates within a few hundred steps. With it, the L_0 pressure spreads the gates across a wide range without driving any sizable fraction all the way to zero, producing a stable ordering of landmark importance that we exploit in §3.3. The L_0 term is multiplied by a piecewise-linear weight schedule $\lambda(t)$ that warms up and then anneals to a target value $\lambda_{\max} = 20$:

$$\lambda(t) = \lambda_{\max} \cdot \text{clip}\left(\frac{t - T_w}{T_a},\, 0,\, 1\right) \tag{4}$$

with warmup length $T_w = 10{,}000$ and anneal length $T_a = 40{,}000$ steps. The warmup gives the encoder time to fit the data on the (nearly) full skeleton before sparsification begins; the anneal then slowly raises the cost of keeping a gate open until the budget meaningfully constrains the gate.

3.3 Selection Strategies

Once the gate has converged, each landmark j has a learned keep probability $p_j = \sigma(\alpha_j)$ that induces

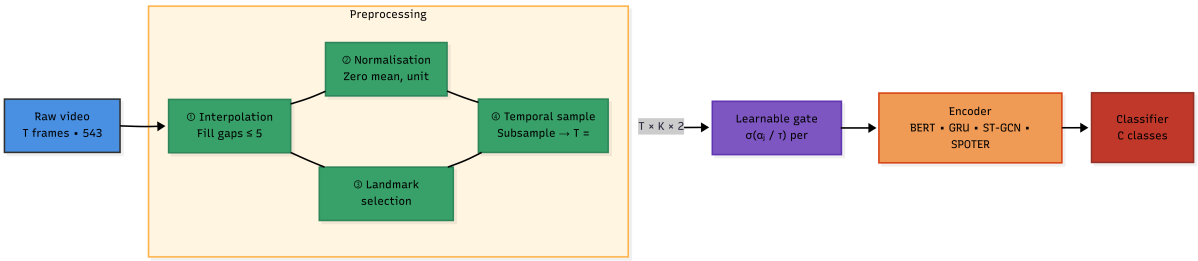

Figure 1: Schematic of the proposed method. Preprocessed skeleton landmarks pass through a learnable per-landmark gate before reaching the encoder. The gate is trained jointly with the encoder via cross-entropy and an L_0 sparsity penalty; the encoder can be any of the four families described in §3.4.

an ordering over the 543 landmarks. We use this ordering to construct fixed landmark subsets at a target size $K \in \{270, 100, 48, 24, 10\}$. We compare two strategies for doing so.

Top-K (one-shot). We rank the 543 landmarks by p_j and select the top K in a single pass. The encoder is then retrained from scratch with the gate replaced by a fixed indexing operation that drops the unselected landmarks. A single L_0 training run thus produces the ordering used at every K, making this the cheapest strategy.

Iterative cascade. Top-K implicitly assumes that the importance ranking learned over the full 543 landmarks is the same ranking we would obtain if the gate could only choose among, say, 48 landmarks. The iterative strategy addresses this assumption. Starting from the full 543, we train the gate, select the top K_1, retrain the gate restricted to those K_1 landmarks, select the top $K_2 < K_1$ from the new ranking, and so on down to the target size. In practice, the cascade descends through five stages with target sizes $K \in \{270, 100, 48, 24, 10\}$, and each stage runs the full L_0 training (the warmup-then-anneal schedule of §3.2) restricted to the previous stage's selected subset. Each stage re-ranks the surviving landmarks under the constraint that only those landmarks are available, so the final K-landmark subset reflects compositional rather than independent landmark importance.

We evaluate both strategies in §5; in many settings iterative gives a noticeable win at small K, but top-K is essentially free given that the L_0 ordering is already produced.

3.4 Skeleton Encoder Family

The gate is encoder-agnostic by design. Any encoder f_θ that consumes a skeleton sequence of shape (T, J, P) and produces a fixed-dimensional embedding $z \in R^d$ can be plugged in behind it. To test the encoder-agnostic claim empirically, we instantiate four encoders spanning the dominant traditions in skeleton-based ISLR: **BERT**, a Transformer that attends over flattened per-frame landmark vectors along the time axis (Vaswani et al., 2017; Devlin et al., 2019); a bidirectional **GRU** (Cho et al., 2014) that processes the same flattened input recurrently; **ST-GCN** (Yan et al., 2018), which imposes an anatomical adjacency over landmarks and applies spatial graph convolutions with temporal convolutions over a fixed frame window; and **SPOTER** (Bohacek and Hruz, 2022), which gives each landmark its own learned embedding and factors spatial and temporal modeling across two Transformers. All four share the contract $(B, T, J, P) \rightarrow (B, d)$ and are exposed through a single factory call, so the gate, optimizer, loss, and data pipeline are identical across encoders. Per-encoder hyperparameters are in Appendix A.

3.5 Training Objective

The full training objective combines cross-entropy classification loss with the scheduled L_0 penalty:

$$\mathcal{L}(\theta, \alpha) \;=\; \mathcal{L}_{\mathrm{CE}}\Big(y,\; f_\theta(\tilde{X})\Big) + \lambda(t)\cdot\mathcal{L}_0(\alpha), \quad (5)$$

where $\tilde{X}$ is the gated skeleton with mask $m_j(\tau(t))$ at training step t, and $\mathcal{L}_{\mathrm{CE}}$ supports optional inverse-frequency class weights for imbalanced datasets. Encoder parameters θ and gate logits α are optimized jointly with AdamW (Loshchilov and Hutter, 2019), distributed over eight GPUs using PyTorch Lightning's DDP strategy (Falcon and The PyTorch Lightning team, 2019). Learning rates and dataset-specific schedule lengths are tuned independently and reported in §4.

Dataset	Classes	Train	Val	Test	Signers
AUTSL	226	28,142	4,418	3,742	43
ASL Citizen	2,731	40,154	10,304	32,941	52
GSL	310	34,957	2,330	3,500	7
Multilingual	3,267	103,253	17,052	40,183	102

Table 1: Dataset summary. Multilingual is the union of the three single-language datasets with disjoint class IDs.

4 Experimental Setup

4.1 Datasets

We use three public isolated sign language datasets, each in a different language, plus a multilingual union of all three. Table 1 lists the splits.

AUTSL (Sincan and Keles, 2020) (Turkish, 226 signs), **ASL Citizen** (Desai et al., 2023) (American, 2,731 signs), and **GSL** (Adaloglou et al., 2020) (Greek, 310 signs) cover three independently-collected sign languages. ASL Citizen uses signer-independent splits (no overlap between train and test signers), which produces the wide validation–test gap we report throughout §5. For multilingual experiments we concatenate the three corpora and remap class labels into a single 3,267-class space, with AUTSL at $[0, 225]$, ASL Citizen at $[226, 2956]$, and GSL at $[2957, 3266]$. Train, validation, and test splits are inherited from each source dataset.

4.2 Experimental Design

Our experiments vary along three axes:

- **Dataset**: each of the four corpora in Table 1.
- **Subset size**: $K \in \{543, 270, 100, 48, 24, 10\}$. $K = 543$ is the full unpruned skeleton.
- **Selection strategy**: top-K or iterative cascade (§3.3). Only meaningful when $K < 543$.

We use BERT (§3.4) as the skeleton encoder throughout. Every point in our Pareto frontiers comes from a two-stage training procedure. Stage 1 trains a BERT encoder with the L_0 gate active over all 543 landmarks; this run produces the landmark ranking. Stage 2 then retrains a fresh BERT encoder on the top-K subset selected from that ranking, with the gate held fixed. All reported test accuracies come from the stage-2 model.

4.3 Implementation Details

Skeleton inputs come from MediaPipe Holistic (Lugaresi et al., 2019), applied frame-by-frame; we keep only the 2D (x, y) coordinates of the 543 returned landmarks, ignoring depth and the per-landmark visibility flag. The four encoders we compare span the dominant skeleton-modeling traditions: BERT and SPOTER are Transformers that attend over landmarks as an unstructured set, the bidirectional GRU processes the skeleton as a recurrent sequence of frames, and ST-GCN imposes an anatomical adjacency on the landmark graph. To ensure that accuracy differences across encoders come from inductive bias rather than capacity, we tune each family independently with Optuna (Akiba et al., 2019) under a shared parameter budget. Per-encoder hyperparameters are in Appendix A; optimizer settings, landmark-pruning hyperparameter values, and random seeds are in Appendix F.

4.4 Baselines

We compare our learned landmark subsets against two skeleton baselines. Each one isolates a different part of the contribution.

The **full-skeleton baseline** ($K = 543$) trains the encoder on all 543 MediaPipe landmarks with no pruning. It is the upper bound on the information any skeletal method has access to, and it appears as the $K = 543$ row of every Pareto frontier we report. The comparison against this baseline answers the question that motivates the paper: can a small selected subset of landmarks match or beat the full skeleton?

The **random-K baseline** uses the same encoder, training schedule, and target subset size as our pruned models, but draws the K landmarks uniformly at random from the 543 rather than from a learned ranking. We perform a single random draw at each $K \in \{270, 100, 48, 24, 10\}$. This fixes the input size and isolates how much of our accuracy comes from which landmarks are kept rather than from how many.

4.5 Evaluation Protocol

We report Top-1 classification accuracy on each dataset's held-out test set. Top-1 is the standard ISLR metric and the one used by the dataset releases we build on. For the multilingual union we also report per-language test accuracy. To produce it, we split the multilingual model's predictions back into AUTSL, ASL Citizen, and GSL by their class-ID ranges. This breakdown lets us see whether multilingual training trades off accuracy on one language for another.

To quantify cross-language landmark agreement

we report the **consensus count** at each K: the number of landmarks selected by all four datasets' stage-1 runs (AUTSL, ASL Citizen, GSL, and the multilingual union). Under independent uniform draws from the 543 landmarks, the expected consensus count is $K^4/543^3$, which is below 1 for every $K \leq 100$; any nonzero observed count therefore reflects agreement well beyond chance.

5 Results

5.1 Pareto Frontier: 24–48 Landmarks Match the Full Skeleton

Figure 2 shows test accuracy as a function of subset size K for each of our four datasets; Table 2 reports the underlying numbers. Aggressive pruning is consistently beneficial: on every dataset, the best pruned configuration matches or exceeds the full-skeleton baseline at $K = 543$, often by a wide margin. AUTSL improves by 7.1 percentage points (iterative, $K = 24$), ASL Citizen by 6.5 points (top-K, $K = 24$), GSL by 2.2 points (top-K, $K = 48$), and the multilingual union by 8.9 points (top-K, $K = 24$). **The most striking finding is the $K = 10$ row.** Less than two percent of MediaPipe Holistic's 543 landmarks suffices to match or beat the full skeleton on three of the four datasets, often substantially. AUTSL's best $K = 10$ configuration reaches 87.36% test accuracy, only 0.5 points shy of its overall best at $K = 24$ and a full 6.6 points above the 543-landmark baseline; ASL Citizen beats its 543 baseline by 4.4 points; and the multilingual union gains 3.3. The sole exception is GSL, which loses 3.4 points at $K = 10$, but that loss is modest given that more than 98% of the input has been discarded. Which ten landmarks survive, and how much overlap there is between languages, is the subject of §5.4. These results show that the bulk of the 543 landmarks are not just redundant but actively detrimental to sign recognition.

5.2 Top-K vs. Iterative Selection

Iterative cascade is the more informed of our two selection strategies by construction. At each stage of the descent, it re-trains the gate on the previous stage's surviving landmarks, giving the model another opportunity to rank those landmarks under the constraint that only they are available. Top-K, by contrast, commits to a single ranking learned over the full 543-landmark candidate set and reuses that ranking at every K. We therefore expect iterative cascade to outperform top-K at aggressive subset sizes, where the relative importance of the surviving landmarks depends most strongly on which other landmarks remain in the input.

Our expectation is partially borne out. At $K = 10$, iterative cascade wins by 4.3 percentage points on AUTSL (87.36% vs 83.08%) and by 3.2 points on ASL Citizen (49.73% vs 46.54%), which is exactly the regime where re-ranking under fewer candidates should help most. On GSL the two strategies are essentially tied at every K. The conspicuous exception is the multilingual union, where top-K beats iterative at every K we tested, by margins as large as 5.2 points at $K = 10$. We do not have a confident explanation for the reversal, but suspect that the multilingual training distribution, which is substantially more heterogeneous than any of the single-language distributions, interacts unfavorably with the cascade's stage-wise re-ranking. The practical takeaway is that iterative selection pays off at aggressive K on monolingual data, while top-K, which is essentially free, is the better default elsewhere.

As a sanity check that the gains come from *which* landmarks are kept rather than just *how many*, we additionally compare both strategies against a random-K baseline at each $K \in \{270, 100, 48, 24, 10\}$ with a single uniform random draw per K found in Table 9.

5.3 Multilingual Recognition

Beyond the per-language Pareto frontiers, we train a single multilingual model on the union of all three datasets with a 3,267-class joint label space, applying the protocol of §4.2 identically. We partition the multilingual model's test predictions back along source-dataset class-ID ranges to recover per-language accuracy.

Table 3 reports test accuracy at $K = 24$, the best subset size for the multilingual model. Compared to the corresponding monolingual model at the same K, the multilingual model is essentially tied on ASL Citizen (51.93% vs 51.78%), loses 2.9 points on AUTSL (84.95% vs 87.89%), and loses 8.0 points on GSL (61.74% vs 69.77%). Multilingual training therefore does not strictly improve recognition for any language we tested, and the largest loss is on GSL, the dataset with the fewest signers. We take this not as a failure of the multilingual setup but as evidence that the single-encoder constraint is a real one: a single model can handle three independently-developed languages

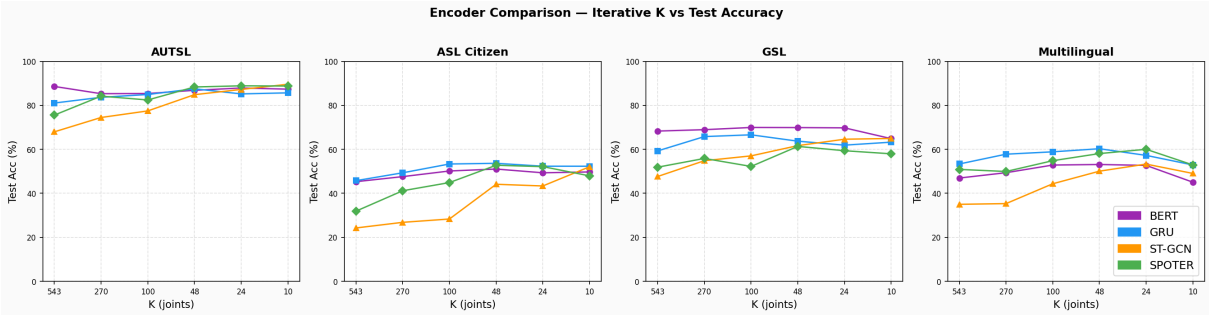

Figure 2: Pareto frontier of Top-1 test accuracy against subset size K for each of the four datasets, with all four skeleton encoders (BERT, GRU, ST-GCN, SPOTER) overlaid. The leftmost point of each curve is the unpruned $K = 543$ baseline. Pruning is beneficial on every dataset, with peaks at $K = 24$–48 and only a modest decline at $K = 10$ (except on GSL). Per-strategy and per-encoder numbers are in Table 2.

	AUTSL		**ASL Citizen**		**GSL**		**Multilingual**	
K	top-K	iter	top-K	iter	top-K	iter	top-K	iter
543	80.76	—	45.30	—	68.31	—	46.96	—
270	85.30	—	47.60	—	68.94	—	49.38	—
100	85.97	85.38	50.32	50.16	67.31	69.94	52.99	52.86
48	86.42	86.83	51.18	51.04	70.54	69.86	55.08	53.11
24	87.41	87.89	51.78	49.34	68.63	69.77	55.86	52.70
10	83.08	87.36	46.54	49.73	64.74	64.86	50.21	45.06

Table 2: Top-1 test accuracy (%) across landmark count K for each dataset, under top-K and iterative cascade selection (BERT encoder for both stages). $K = 543$ is the unpruned baseline; iterative is undefined at $K = 543$. Best per-dataset accuracy is reached at $K \in \{24, 48\}$ in every case.

Language	**Mono.**	**Multi.**	Δ
AUTSL	87.89	84.95	-2.94
ASL Citizen	51.78	51.93	$+0.15$
GSL	69.77	61.74	-8.03
Combined (3,267 cls.)	—	55.86	—

Table 3: Per-language test accuracy (%) at $K = 24$ under monolingual vs. multilingual training. Monolingual numbers are the best-strategy result for each dataset at $K = 24$ from Table 2; multilingual numbers come from partitioning the multilingual model's predictions back along source-dataset class-ID ranges.

with reasonable accuracy, but per-language fine-tuning still wins. The stronger evidence for cross-language convergence comes from the landmark-overlap analysis in §5.4.

5.4 Cross-Language Landmark Overlap

For the language-agnostic claim to hold, the landmark subsets chosen monolingually for each of our three sign languages should overlap substantially, and the anatomical regions they cover should be similar. Figure 3 reports both at once: the 543 landmarks are plotted on a skeleton silhouette at each K, and each landmark is colored by how many of the four datasets (AUTSL, ASL Citizen, GSL, and the multilingual union) independently selected it at stage 1.

The dominant impression is one of substantial agreement. At every K we tested, a clear majority of the selected landmarks are picked by multiple languages. A non-trivial subset is picked by all four: 3 of 10 at $K = 10$, 11 of 24 at $K = 24$, 30 of 48 at $K = 48$, 67 of 100 at $K = 100$, and 158 of 270 at $K = 270$. Random draws of K-subsets from the 543 would share only a tiny fraction of landmarks by chance, yet the learned subsets share a clear plurality. We read this as direct empirical evidence that the landmarks carrying signal for sign-language recognition are largely the same regardless of which sign language is being recognized.

This same figure also confirms the linguistic predictions in §2: lexical content is carried primarily on the hands, the upper body positions them, and non-manual markers on the face encode grammatical rather than lexical meaning. The manual articulators dominate at every K: the hands are heavily over-represented relative to their raw proportion of the 543 landmarks. The face is the first region to drop out as K shrinks, consistent with non-manuals carrying grammatical rather than lexical contrasts, while hand landmarks persist down to the smallest budgets we tested. Body landmarks occupy an intermediate niche, providing the anchoring infor-

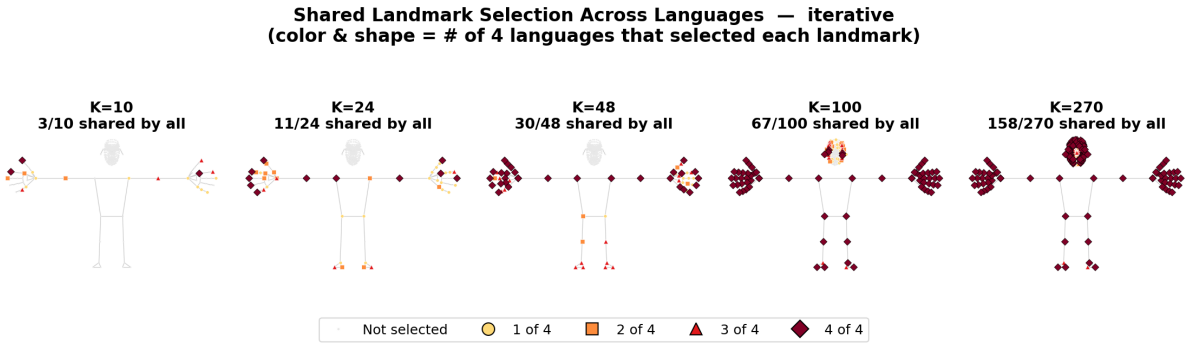

Figure 3: Shared landmark selection across languages. For each K, a single skeleton silhouette plots the 543 MediaPipe landmarks; each dot is colored by the number of the four datasets (AUTSL, ASL Citizen, GSL, Multilingual) whose stage-1 L_0 run independently selected that landmark. Dark cells mark landmarks selected by all four; gray dots are landmarks that were not selected at this K. The visualization combines spatial location, body-part composition, and cross-language agreement in one image.

Encoder	AUTSL	ASL Cit.	GSL	Mult.
BERT	87.89	51.78	70.54	55.86
GRU	87.5	54.3	66.6	60.5
ST-GCN	89.6	52.1	67.8	55.2
SPOTER	89.8	52.8	61.3	60.7

Table 4: Encoder comparison at each language's best K (top-K or iterative). All encoders are retrained on the BERT-derived landmark subset. BERT numbers are reproduced from Table 2.

mation that situates the hands in space. Precise per-K counts are reported in Appendix B.

5.5 Encoder Comparison

Beyond the per-language Pareto frontiers, we run a stage-2 encoder comparison: holding the BERT-derived landmark subset fixed at each K, we retrain BERT, GRU, ST-GCN, and SPOTER on that subset. Two patterns in Table 4 (full per-encoder grids in Appendix C) make a stronger input-property argument than curve similarity alone. First, non-Transformer encoders gain far more from pruning than BERT: going from K=543 to each encoder's best K, BERT gains only 2–9 points across the four datasets, while ST-GCN gains 20–28 and SPOTER 9–21. The 468-landmark face block does not merely fail to help these architectures; it actively prevents them from competing with BERT. Second, the encoders converge once pruned: cross-encoder spread at K=543 is 13–22 points per dataset, and at each dataset's best K that spread collapses to 2–9 points. Together these say that accuracy at an informative landmark budget is determined more by which landmarks are kept than by which encoder reads them.

6 Conclusion

We asked how few of MediaPipe Holistic's 543 landmarks an isolated sign-language recognizer actually needs. Using a learnable L_0-regularized gate to rank the landmarks and retraining on the top-K subset, we find that 24 to 48 landmarks suffice to match or exceed the full skeleton on three typologically unrelated sign languages: AUTSL (Turkish), ASL Citizen (American), and GSL (Greek), as well as on a multilingual union of all three. Even at $K = 10$, less than two percent of the 543, accuracy remains competitive with the full skeleton on three of the four datasets, with only a modest loss on GSL; this is far more aggressive sparsification than any prior manual reduction has reported, and it sharpens our interpretation that the articulators carrying the bulk of lexical content are captured by a budget on the order of the four phonological parameters themselves. The same broad landmark subset emerges across all three sign languages despite their independent origins, which is evidence that the informative skeletal subset is a property of sign language as a linguistic system rather than of any particular sign language.

One observation from our experiments is worth highlighting: the L_0 gate does not converge to a binary $\{0, 1\}$ regime. What it produces, reliably, is a clear ranking of landmark importance, and the discrete subset comes from the downstream retraining stage.

Limitations

We evaluate on three sign languages and one multilingual union; the broader claim that the informative subset is shared across all sign languages

would require evaluation on a substantially larger and more typologically diverse set of languages than the three we use here. We use only the 2D coordinates returned by MediaPipe Holistic, so we cannot say whether the same subset emerges from 3D landmarks, depth, or a different pose extractor. Stage-1 landmark selection is run with BERT only; whether other encoder architectures would select substantially different subsets at stage 1 is left to future work. Our reported Pareto-frontier numbers come from a single random seed; multi-seed variance estimates are not yet available for all configurations. Finally, we defer cross-language transfer experiments, in which one language's learned subset is used to train a model in another language, to future work.

Acknowledgments

In accordance with the ACL Policy on Publication Ethics, we disclose that a generative AI assistant (Claude) was used in preparing this paper. The assistant was used for prose editing, LaTeX formatting, and proofreading of the manuscript, and for generating scaffolding for portions of the data-loading pipeline, training loop, and analysis scripts. All research contributions (problem framing, the L_0 landmark-pruning method, experimental design, every experiment reported, and the analysis and interpretation of results) are the authors' own. Code produced with assistance was reviewed, modified, and integrated by the authors, who take responsibility for its correctness.

Ethical considerations

This work is motivated by accessibility for Deaf and hard-of-hearing communities, for whom sign language technology shapes access to information, services, and education. Sign languages are full natural languages with their own phonology and grammar, and we frame the work accordingly. We build on a community-grounded benchmark, ASL Citizen (Desai et al., 2023), that was crowdsourced from Deaf signers in their home environments, and we use the official splits as released. We acknowledge that our authorship is not Deaf-led, and we treat this paper as one technical contribution within a larger accessibility research agenda that should center Deaf perspectives.

The three datasets we use (AUTSL (Sincan and Keles, 2020), ASL Citizen (Desai et al., 2023), and the VCL GSL corpus (Adaloglou et al., 2020)) are all public academic releases used per their license terms, and we add no new data collection. Because the model consumes sparse 2D landmark coordinates rather than raw video frames, the input representation abstracts away most personally identifying detail beyond what the source datasets release. Our system performs isolated single-sign classification rather than translation, and treating its outputs as a translation pipeline risks misrepresenting signers' meaning in deployed contexts; we caution against substituting ISLR artifacts for qualified human interpreters in medical, legal, or educational settings where misrecognition can cause real harm.

References

Nikolas Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Th Papadopoulos, Vassia Zacharopoulou, George J Xydopoulos, Klimnis Atzakas, Dimitris Papazachariou, and Petros Daras. 2020. A comprehensive study on sign language recognition methods. *arXiv preprint arXiv:2007.12530*.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. *Preprint*, arXiv:1907.10902.

Robbin Murray Battison. 1977. *Lexical Borrowing in American Sign Language: Phonological and Morphological Restructuring*. ProQuest Dissertations and Theses, University of California, San Diego.

Matyas Bohacek and Marek Hruz. 2022. Sign Pose-based Transformer for Word-level Sign Language Recognition .

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *Preprint*, arXiv:1406.1078.

Mathieu De Coster, Mieke Van Herreweghe, and Joni Dambre. 2021. Isolated sign recognition from rgb video using pose flow and self-attention. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3436–3445.

Aashaka Desai, Lauren Berger, Fyodor O. Minakov, Vanessa Milan, Chinmay Singh, Kriston Pumphrey, Richard E. Ladner, Hal Daumé III, Alex X. Lu, Naomi Caselli, and Danielle Bragg. 2023. Asl citizen: A community-sourced dataset for advancing isolated sign language recognition. *Preprint*, arXiv:2304.05934.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

William Falcon and The PyTorch Lightning team. 2019. PyTorch Lightning.

Mo Guan, Yan Wang, Guangkun Ma, Jiarui Liu, and Mingzu Sun. 2024. Multi-stream keypoint attention network for sign language recognition and translation. *Preprint*, arXiv:2405.05672.

Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li, and Yun Fu. 2021. Skeleton aware multi-modal sign language recognition. *Preprint*, arXiv:2103.08833.

Youngmin Kim and Hyeongboo Baek. 2023. Preprocessing for keypoint-based sign language translation without glosses. *Sensors*, 23(6).

Dongxu Li, Cristian Rodriguez Opazo, Xin Yu, and Hongdong Li. 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. *Preprint*, arXiv:1910.11006.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Christos Louizos, Max Welling, and Diederik P. Kingma. 2018. Learning sparse neural networks through l_0 regularization. *Preprint*, arXiv:1712.01312.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. Mediapipe: A framework for perceiving and processing reality. In *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019*.

Wendy Sandler. 2012. The phonological organization of sign languages. *Lang Linguist Compass*.

Prem Selvaraj, Gokul NC, Pratyush Kumar, and Mitesh Khapra. 2021. Openhands: Making sign language recognition accessible with pose-based pretrained models across languages. *Preprint*, arXiv:2110.05877.

Ozge Mercanoglu Sincan and Hacer Yalim Keles. 2020. Autsl: A large scale multi-modal turkish sign language dataset and baseline methods. *IEEE Access*, 8:181340–181355.

Jr. Stokoe, William C. 2005. Sign language structure: An outline of the visual communication systems of the american deaf. *The Journal of Deaf Studies and Deaf Education*, 10(1):3–37.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Preprint*, arXiv:1706.03762.

Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. 2020. Deep high-resolution representation learning for visual recognition. *Preprint*, arXiv:1908.07919.

Fangyun Wei and Yutong Chen. 2023. Improving continuous sign language recognition with cross-lingual signs. *Preprint*, arXiv:2308.10809.

Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. *Preprint*, arXiv:1801.07455.

Aoxiong Yin, Zhou Zhao, Weike Jin, Meng Zhang, Xingshan Zeng, and Xiaofei He. 2022. Mlslt: Towards multilingual sign language translation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5099–5109.

Ulrike Zeshan. 2004. Interrogative constructions in signed languages: Crosslinguistic perspectives. *Language*.

A Encoder Hyperparameters

Table 5 lists the per-encoder hyperparameters used in all experiments. Each encoder family is tuned independently with Optuna under a shared trainable-parameter-count budget, so accuracy differences across encoders reflect differences in inductive bias rather than capacity.

BERT		**GRU**	
Hidden dim	512	Hidden dim	512
Layers	2	Layers	2
Attention heads	8	Bidirectional	yes
Intermediate size	2048	Dropout	0.1
Dropout	0.1	Params (M)	8.45
Params (M)	6.89		

ST-GCN		**SPOTER**	
Channels c_1–c_4	64–512	Hidden dim	512
Stages	4	Spatial layers	2
Temporal kernel	13	Temporal layers	2
Dropout	0.0	Attention heads	8
Params (M)	5.14	Dropout	0.1
		Params (M)	7.13

Table 5: Per-encoder hyperparameters, tuned independently with Optuna under a shared parameter-count budget. For ST-GCN, channel widths double at each stage: c_1=64, c_2=128, c_3=256, c_4=512.

B Body-Part Composition of Selected Subsets

Figure 4 reports the body-part composition of the landmark subsets selected on each dataset at each K, broken down by MediaPipe Holistic's four anatomical groups (face, body, left hand, right hand). The headline patterns discussed in §5.4 are visible directly: the hands dominate at every K, the face is the first region to drop out as K shrinks, and body landmarks occupy an intermediate niche.

C Per-Encoder Pareto Tables

These tables expand the BERT Pareto grid of Table 2 to the other three encoder families. Each table follows the same layout: rows are subset sizes K, columns are (top-K, iterative) pairs per dataset. BERT numbers are reproduced from Table 2 for cross-encoder reference.

	AUTSL		ASL Cit.		GSL		Mult.	
K	topK	iter	topK	iter	topK	iter	topK	iter
543	81.05	—	45.76	—	59.29	—	53.35	—
270	—	83.65	—	49.34	—	65.80	—	57.80
100	83.19	84.93	52.74	53.34	65.97	66.60	59.48	58.87
48	84.71	87.52	52.80	53.65	62.77	63.71	60.47	60.28
24	84.71	85.20	54.28	52.33	66.20	61.91	60.31	57.25
10	82.92	85.68	48.57	52.32	64.46	63.26	53.89	52.90

Table 6: **GRU** encoder Pareto frontier.

	AUTSL		ASL Cit.		GSL		Mult.	
K	topK	iter	topK	iter	topK	iter	topK	iter
543	67.96	—	24.26	—	47.66	—	35.01	—
270	—	74.48	—	26.82	—	54.83	—	35.33
100	79.10	77.47	31.98	28.33	55.69	57.03	41.42	44.35
48	87.63	84.85	37.58	44.12	62.60	61.69	46.93	50.08
24	84.50	87.23	47.12	43.35	59.26	64.60	55.18	53.29
10	83.32	89.58	47.73	52.07	67.83	64.94	53.73	49.05

Table 7: **ST-GCN** encoder Pareto frontier.

	AUTSL		ASL Cit.		GSL		Mult.	
K	topK	iter	topK	iter	topK	iter	topK	iter
543	75.57	—	31.91	—	51.89	—	50.88	—
270	—	84.18	—	41.18	—	55.86	—	49.93
100	84.95	82.44	45.54	44.89	52.83	52.23	56.22	54.84
48	89.79	88.35	50.06	52.76	60.89	61.34	56.71	58.08
24	89.71	88.88	49.67	52.14	60.60	59.37	60.71	60.02
10	87.01	88.78	47.56	47.95	59.37	57.97	54.85	52.91

Table 8: **SPOTER** encoder Pareto frontier.

D Random-K Baseline

The random-K baseline holds the input budget fixed at each $K \in \{270, 100, 48, 24, 10\}$ but draws the retained landmarks uniformly at random from the 543. We use a single random draw per K (draw 0) to remain consistent across experiments. Each encoder is retrained from scratch on its random subset. Table 9 reports the resulting test accuracies for direct comparison with the learned-selection rows of Table 2 and Appendix C.

Encoder	K	AUTSL	ASL	GSL	Mult.
BERT	270	75.07	51.8	64.7	47.4
BERT	100	74.59	49.3	64.8	45.3
BERT	48	43.16	28.3	50.5	23.4
BERT	24	67.56	9.6	29.6	7.2
BERT	10	1.74	0.1	1.5	0.3
GRU	270	80.06	47.52	63.03	51.95
GRU	100	76.75	45.63	63.23	51.39
GRU	48	41.72	28.14	57.09	0.42
GRU	24	71.11	9.89	31.20	18.16
GRU	10	13.42	1.08	9.14	2.73
ST-GCN	270	76.38	29.25	50.69	38.3
ST-GCN	100	75.28	26.85	55.57	32.83
ST-GCN	48	46.53	20.17	53.46	26.70
ST-GCN	24	69.03	5.76	33.17	8.71
ST-GCN	10	16.14	1.21	15.97	3.55
SPOTER	270	83.19	37.08	53.20	46.5
SPOTER	100	78.67	38.32	58.40	37.59
SPOTER	48	40.57	22.06	48.03	27.65
SPOTER	24	70.07	6.11	30.37	13.95
SPOTER	10	12.59	1.03	8.89	2.21

Table 9: Random-K baseline test accuracy (%) under a single uniform random draw (draw 0) of K landmarks from the 543, broken down by encoder and dataset. The GRU multilingual run at $K = 48$ did not converge.

E Full Multilingual Breakdown

Table 10 expands the compact $K = 24$ comparison of Table 3 to the full Pareto frontier under both selection strategies. Each row reports overall multilingual test accuracy and the per-language partitions for AUTSL, ASL Citizen, and GSL recovered by splitting predictions along source-dataset class-ID ranges.

K	Strategy	Overall	AUTSL	ASL	GSL
543	top-K	46.96	80.01	42.44	54.17
270	top-K	49.38	80.14	45.48	53.23
100	top-K	52.99	83.22	48.91	59.23
100	iterative	52.86	81.80	49.28	55.57
48	top-K	55.08	83.24	51.50	58.54
48	iterative	53.11	82.84	48.66	63.26
24	top-K	55.86	84.95	51.93	61.74
24	iterative	52.70	83.00	48.33	61.37
10	top-K	50.21	73.14	46.66	59.03
10	iterative	45.06	81.88	42.23	58.46

Table 10: Multilingual model test accuracy (%) at every K under top-K and iterative cascade selection, with per-language partitions. Overall accuracy is the unweighted Top-1 over the 40,183 multilingual test clips.

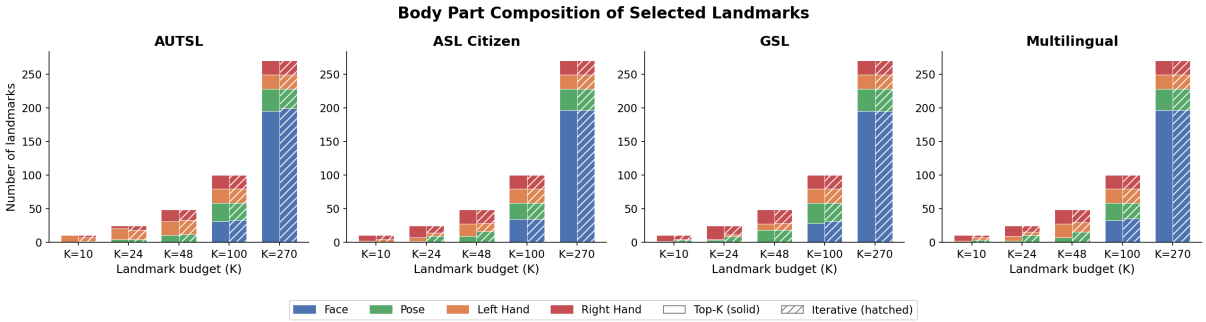

Figure 4: Body-part composition of the selected landmark subset at each K, broken down by MediaPipe Holistic's four anatomical groups. Hands dominate at every K; face landmarks effectively vanish below $K = 100$.

F Training Details

Optimization. We use AdamW with weight decay 0.01, an effective batch size of 64, and gradient norm clipping at 1.0. Training runs on eight GPUs of a single node with PyTorch Lightning's DDP strategy. The cross-entropy loss is weighted by inverse class frequency, which matters most on ASL Citizen and on the multilingual union. Each run trains for up to 200,000 steps. We validate four times per epoch and stop training early if validation accuracy fails to improve for 30 consecutive checks. The best-validation checkpoint is the one we test.

Landmark-pruning hyperparameters. The schedule constants from §3.2 are held fixed across every (dataset, encoder, K) triple: $p_0 = 0.98$, $\tau_0 = 10$, $\tau_{\text{end}} = 0.01$, $T_\tau = 50{,}000$, $T_w = 10{,}000$, $T_a = 40{,}000$, and $\lambda_{\max} = 20$. With these fixed, the only things changing across experiments are the input space (which landmarks, and how many) and the encoder architecture. The optimization regime that the gate operates under is the same everywhere.

Random seeds. All reported results use a single random seed (`seed = 4000`) unless otherwise noted.

G Landmark Probability Evolution

Figure 5 visualizes the central claim of §3.2: that the L_0 gate produces a clear ordering of landmark importance rather than converging to binary $\{0, 1\}$ decisions. Each curve tracks one landmark's learned probability $\sigma(\alpha_j)$ over training; the curves spread across a wide range as the schedule progresses, but only a small fraction reach 0. The discrete subset reported throughout the paper is constructed by ranking these learned probabilities and re-training on the top K, not by reading off the gate's hard thresholds.

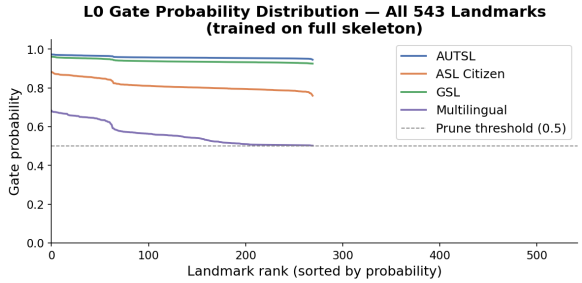

Figure 5: Learned landmark-keep probabilities $\sigma(\alpha_j)$ over training steps. The gate spreads probabilities across a wide range without collapsing any sizable fraction to zero, justifying the ranking-based selection used by both top-K and iterative strategies.

H Compute Resources

All experiments were trained on a single node with eight A-100 GPUs distributed via PyTorch Lightning's DDP strategy; testing uses a single GPU. Each training run targets up to 200,000 steps but is typically stopped earlier by the 30-validation-check early-stopping criterion. Stage-1 L_0 runs follow the same schedule.

Wall-clock per training run. Median wall-clock per training run, estimated from 355 SLURM job logs, is approximately 10 hours on AUTSL, 17 hours on ASL Citizen, 5 hours on GSL, and 22 hours on the multilingual union, with substantial variance driven by encoder size and dataset volume.

Per-encoder totals. Summed GPU-hours (wall-clock $\times$ 8 GPUs) for training across all configurations: BERT $\approx$ 1,300 GPU-h; GRU $\approx$ 1,340 GPU-h; ST-GCN $\approx$ 1,250 GPU-h; SPOTER $\approx$ 1,550 GPU-h. Test-time evaluation across all configura-

tions adds a further $\sim$30 GPU-hours (1-GPU jobs, $<$ 10 minutes each).

I Reproducibility Checklist

Data, Licensing, and Ethics. All three datasets are publicly available: AUTSL (Sincan and Keles, 2020), ASL Citizen (Desai et al., 2023), and the VCL Greek Sign Language corpus (Adaloglou et al., 2020). We use the official train, validation, and test splits provided by each release. Class-label mappings into the multilingual 3,267-class space are deterministic and given in §4.

All existing artifacts are utilized strictly in accordance with their intended research purposes and licenses: AUTSL is released for academic research purposes (Sincan and Keles, 2020); ASL Citizen is licensed under the Microsoft Research License Agreement for non-commercial research (Desai et al., 2023); and GSL is a public benchmark for academic evaluation (Adaloglou et al., 2020).

Regarding privacy and content moderation, we did not collect new data. The source datasets consist of public sign videos. Furthermore, because our method extracts and processes sparse 2D landmark coordinates (MediaPipe Holistic) rather than raw RGB video frames, the input representation inherently abstracts away personally identifying information and signer identity.

Models and hyperparameters. Encoder hyperparameters are listed in Table 5; landmark-pruning hyperparameters, optimizer settings, batch sizes, and the early-stopping criterion are listed in §F. All schedule constants are held fixed across (dataset, encoder, K) triples and were not retuned per-condition.

Hyperparameter search. Per-encoder hyperparameters were tuned independently with Optuna (Akiba et al., 2019) under a shared trainable-parameter budget to make encoders comparable; landmark-pruning hyperparameters were chosen once and fixed.

Evaluation. We report Top-1 classification accuracy on each dataset's held-out test set. Best-of-validation-accuracy checkpoint selection is used for testing. For the multilingual model, per-language accuracy is recovered by partitioning predictions along source-dataset class-ID ranges.

Random seeds. All results use `seed = 4000` unless otherwise noted; multi-seed variance estimates are not yet available.

Compute. See Appendix H.

Code and data release. Source code, configuration files, and learned landmark-index JSONs will be released upon publication.