Knowledge Distillation: A Visual Guide

Roy Miles

March 2026

Abstract

Knowledge distillation trains a compact student model with supervision from a stronger teacher. This note gives a conference-style overview of the main forms of distillation: response matching at the output layer, representation matching at intermediate layers, and relation matching over pairs or groups of samples. It also summarizes projector-based feature distillation, common loss functions, recent language-model variants, and several open directions.

1 Teacher–Student Framework

The central idea of knowledge distillation (KD) is to use a trained, high-capacity teacher network to guide a smaller student. Instead of learning only from hard labels, the student is also encouraged to reproduce information contained in the teacher's predictions, hidden states, or relational geometry. Hinton et al. [1] popularized the output-level version of this idea by showing that softened class probabilities expose useful "dark knowledge" about inter-class similarity.

Modern surveys often group KD objectives according to the signal transferred from teacher to student [2]:

- **Response-based distillation** aligns final logits or probability distributions.
- **Feature-based distillation** aligns internal representations, usually with an adapter or projector.
- **Relation-based distillation** aligns the structure among samples, layers, or embeddings.

These families are not mutually exclusive. Many practical recipes combine a task loss with one or more distillation losses, using the teacher as an additional training signal rather than as a replacement for labels.

Takeaway. KD is most useful when the teacher exposes structure that labels alone do not: relative class probabilities, intermediate visual features, sample-to-sample geometry, reasoning traces, or denoising trajectories.

2 Response-Based Distillation

Response-based KD is the classic formulation. Let z_i denote a logit for class i. A temperature parameter T softens the predictive distribution:

$$p_i(T) = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)}. \tag{1}$$

For $T = 1$, this is the ordinary softmax. For larger T, class probabilities become less peaked, so the student sees how the teacher ranks non-target classes. A common objective combines cross-entropy on labels with a Kullback–Leibler term on softened probabilities:

$$\mathcal{L} = (1-\alpha)\,\mathcal{L}_{\mathrm{CE}}(y, p_S) + \alpha T^2\,\mathrm{KL}(p_T(T)\,\|\,p_S(T))\,. \tag{2}$$

The multiplier T^2 is commonly used to keep gradients at high temperature on a comparable scale.

Variants differ mainly in how the teacher distribution is produced or decomposed. Born-Again Networks repeatedly train a new model from a previous generation of the same architecture [3]. Self-distillation builds the teacher signal inside a single network, often through auxiliary heads or deeper layers. Label smoothing may be interpreted as using a simple uniform teacher. Decoupled Knowledge Distillation separates target-class and non-target-class terms, allowing the dark-knowledge component to be weighted more directly [14].

3 Feature-Based Distillation

Output distributions can hide useful information already compressed away by the classifier head. Feature-based KD therefore aligns hidden activations, attention maps, or layer statistics. In convolutional networks, such features often encode a visual hierarchy from edges and textures to object parts. In transformer architectures, the analogous signals may be token embeddings, attention maps, or selected hidden states.

3.1 Projectors and Adapters

A recurring problem is dimensional mismatch. A teacher tensor $f_T \in \mathbb{R}^{C_T \times H \times W}$ and a student tensor $f_S \in \mathbb{R}^{C_S \times H \times W}$ cannot be compared coordinate-wise when $C_T \neq C_S$. A lightweight projector φ maps one representation into the other's space:

$$\mathcal{L}_{\mathrm{feat}} = \|f_T - \varphi(f_S)\|_2^2\,. \tag{3}$$

The projector is optimized during training and usually removed at inference, so it need not affect deployment cost. FitNets used this idea to train thin students from intermediate teacher hints [4]. Later work showed that the projector is not merely a shape converter; it can also change the geometry of the gradient signal [15].

3.2 Projector Dynamics

For a batch of student representations $Z_s \in \mathbb{R}^{B \times d_s}$ and teacher representations $Z_t \in \mathbb{R}^{B \times d_t}$, consider a bias-free linear projec-

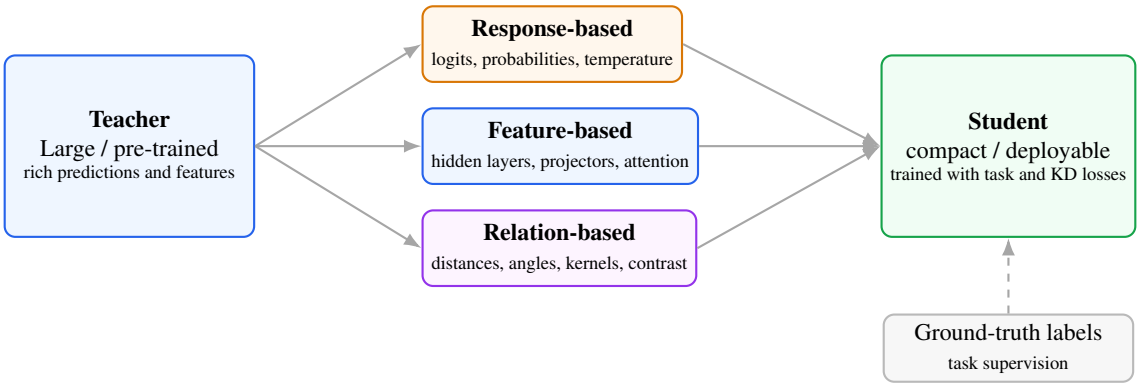

Figure 1: A high-level view of the three main KD paradigms. Response losses operate on output distributions, feature losses act on internal representations, and relation losses match the geometry induced by the teacher.

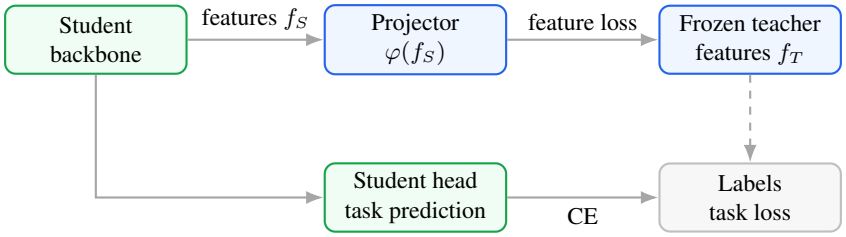

Figure 2: A typical feature-distillation pipeline. The student is trained both for the original task and to match teacher features through a projector. The projector can be discarded after training.

tor $W_p \in \mathbb{R}^{d_s \times d_t}$ and squared discrepancy

$$D(Z_s, Z_t; W_p) = \frac{1}{2} \|Z_s W_p - Z_t\|_F^2 . \tag{4}$$

The negative gradient with respect to W_p is

$$\dot{W}_p = -\frac{\partial D}{\partial W_p} = -Z_s^\top Z_s W_p + Z_s^\top Z_t = C_{st} - C_s W_p, \tag{5}$$

where $C_s = Z_s^\top Z_s$ is the student self-correlation matrix and $C_{st} = Z_s^\top Z_t$ is the student–teacher cross-correlation. This makes the projector a compact store of relational information rather than a passive adapter. At stationarity,

$$C_{st} - C_s W_p = 0. \tag{6}$$

If student features are whitened or decorrelated so that $C_s \approx I$, then

$$C_s = I \quad \Rightarrow \quad W_p = C_{st}. \tag{7}$$

Thus normalization and projection are coupled: the normalization scheme controls the geometry in which the projector encodes cross-relations.

3.3 Cross-Architecture Transfer

Distilling across architecture families is harder than distilling between models of the same type. A convolutional teacher and transformer student may organize information very differently, and a high-capacity projector can absorb the alignment burden while leaving the student backbone under-trained. VkD addresses this by constraining the projector to preserve structure [19]. If a kernel can be expanded as

$$k(Z_s^i, Z_s^j) = \sum_{n=0}^{\infty} a_n \langle Z_s^i, Z_s^j \rangle^n, \tag{8}$$

then preserving the kernel reduces to preserving inner products. For a linear projection P, this motivates

$$\langle Z_s^i, Z_s^j \rangle = \langle Z_s^i P, Z_s^j P \rangle, \qquad PP^\top = I, \tag{9}$$

for the row-orthogonal case. When the singular values are constrained, the projector cannot freely collapse directions just to reduce loss; more useful adaptation must happen in the student itself.

3.4 Hints, Attention, and Dense Prediction

FitNets introduced a two-stage recipe: first train the student to reproduce a teacher hint layer, then fine-tune with task and distillation losses [4]. Attention Transfer instead matches spatial attention summaries:

$$A(F) = \sum_c |F_c|^2 \in \mathbb{R}^{H \times W}, \tag{10}$$

$$\mathcal{L}_{\text{AT}} = \sum_l \left\| \frac{A(F_T^l)}{\|A(F_T^l)\|} - \frac{A(F_S^l)}{\|A(F_S^l)\|} \right\|_2^2 . \tag{11}$$

Because attention maps discard the channel basis, they can be easier to transfer across dissimilar networks [5]. ReviewKD

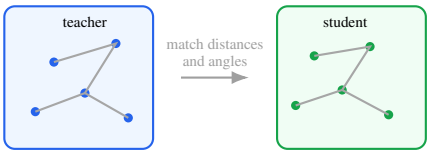

Figure 3: Relation-based KD transfers the shape of the embedding space rather than individual feature coordinates.

generalizes layer matching by allowing a student layer to consult multiple teacher levels [28], while feature-overhaul methods modify the feature loss to avoid harmful gradients in inactive regions [30].

Feature-based KD also appears in cross-task and dense-prediction settings. Cross-task distillation transfers representations from, for example, a classification teacher to a segmentation or depth student through a learned bridge [22]. Dense tasks such as detection, tracking, segmentation, and video object segmentation require supervision at spatial or spatio-temporal positions rather than only at a global embedding [23].

4 Relation-Based Distillation

Relation-based methods ask the student to match geometry, not coordinates. Even if the student cannot reproduce exact teacher features, it may still learn the teacher's distances, angles, neighbourhoods, or distributional structure.

Examples include Relational KD, which matches pairwise distances and angles [6]; Contrastive Representation Distillation, which treats teacher and student views of the same sample as positives [7]; Neuron Selectivity Transfer, which matches activation distributions using MMD [8]; Probabilistic Knowledge Transfer, which matches probability distributions induced by feature-space kernels [31]; and SSKD, which adds self-supervised signals to supervised distillation [29].

Projector-based feature KD provides an interesting bridge to relation-based objectives. Equation (5) shows that the projector dynamics depend on $C_{st} = Z_s^\top Z_t$, a relational quantity over a batch. This helps explain why a well-designed feature loss with a learned projector can sometimes rival more explicit relation losses.

5 Loss Functions

Most KD systems optimize a weighted mixture:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}} + \lambda_{\text{resp}}\mathcal{L}_{\text{response}} + \lambda_{\text{feat}}\mathcal{L}_{\text{feature}} + \lambda_{\text{rel}}\mathcal{L}_{\text{relation}}.$$

The components should be chosen to match the available teacher signal and the deployment constraints of the student.

The basic squared feature loss is simple but sensitive to scale and covariance. Cosine alignment focuses on direction, which can stabilize training but discards norm information. Soft-maximum losses emphasize larger mismatches without forcing every coordinate to be matched equally [15]. Gram, MMD, and contrastive losses move the supervision from coordinates to geometry.

Table 1: Common losses used in KD.

Loss	Typical use
$\mathrm{KL}(p_T \Vert p_S)$	Response matching with soft labels.
$\Vert f_T - \varphi(f_S)\Vert_2^2$	Direct feature or hint matching.
$1 - \cos(f_T, \varphi(f_S))$	Directional feature alignment when norms are unreliable.
$\log \sum_i \vert Z_s W_p - Z_t \vert_i^\alpha$	Soft-maximum feature loss for large capacity gaps.
NCE / contrastive	Relation-based alignment using positive and negative pairs.
Gram / MMD	Distribution or kernel matching over activations.

6 Method Comparison

Feature-based methods often give large accuracy gains but introduce design choices around which layers to match, how to normalize, and how to choose the projector. Relation-based methods can be more architecture-agnostic but may require batch-level computation or memory banks. Response-based KD is usually the easiest to implement but may miss information stored in intermediate representations.

7 Language vs. Vision

Vision and language use the same high-level KD ideas but different training signals. Vision models commonly distill logits, spatial maps, patch tokens, or dense features. Language models distill distributions over very large vocabularies, generated sequences, attention relations, hidden states, or full reasoning traces.

Sequence-level KD trains a student on teacher-generated outputs instead of only on reference labels [12]. Later work such as Distilling Step-by-Step and SCOTT shows that teacher rationales can become useful supervision for smaller reasoning models [16, 17]. Feature distillation is more delicate in language because token positions, sequence lengths, attention heads, and tokenizers may not align. DistilBERT, MiniLM, and TinyBERT illustrate several practical forms of transformer distillation [9, 10, 11].

Tokenizer mismatch is a central LLM issue: if teacher and student vocabularies differ, token-level KL is not directly defined. Alternatives include vocabulary projection, sentence-level alignment, hidden-state matching, or distilling decoded traces. Speculative decoding is related but distinct: a small draft model proposes tokens while a larger verifier preserves the target distribution during inference [18].

7.1 Distillation Tokens in DeiT

DeiT inserts a learnable distillation token into the Vision Transformer sequence [20]. Patch tokens, the class token, and the distillation token are processed together by self-attention. The class head uses the usual supervised objective, while the distillation head is trained from a teacher prediction:

$$\mathcal{L} = \tfrac{1}{2}\mathcal{L}_{\text{CE}}(y, \hat{y}_{\text{cls}}) + \tfrac{1}{2}\mathcal{L}_{\text{KL}}(\hat{y}_{\text{teacher}}, \hat{y}_{\text{dist}}). \qquad (12)$$

Table 2: Representative distillation methods.

Method	Paradigm	Transferred signal	Main constraint	Reference
KD	Response	Soft logits with temperature	Same output space	Hinton et al. [1]
DKD	Response	Target and non-target logit terms	Same output space	Zhao and Cui [14]
FitNets	Feature	Mid-layer hints through a projector	Width-aligned feature spaces	Romero et al. [4]
AT	Feature	Spatial attention maps	Matching or rescaled spatial maps	Zagoruyko and Komodakis [5]
Projector recipe	Feature	Projected features and implicit relations	Adapter and normalization choices	Miles and Mikolajczyk [15]
VkD	Feature	Orthogonally projected features	Row-orthogonal projection	Miles et al. [19]
ReviewKD	Feature	Multi-level teacher representations	Cross-layer aggregation	Chen et al. [28]
RKD	Relation	Pairwise distances and angles	Batch-level relation computation	Park et al. [6]
CRD	Relation	Contrastive teacher–student views	Negatives or memory bank	Tian et al. [7]
DeiT	Response/token	Teacher signal through a distillation token	Vision-transformer architecture	Touvron et al. [20]

This makes DeiT a hybrid: the loss is output-level, but the supervision is routed through an internal token that interacts with the whole image representation.

7.2 Reasoning Distillation

DeepSeek-R1 demonstrates a process-level form of distillation for language reasoning [21]. A large reinforcement-trained teacher generates long reasoning traces, and smaller models are fine-tuned on those traces using ordinary cross-entropy. The transferred object is not only an output distribution but a problem-solving trajectory: exploration, revision, intermediate claims, and final answer formation. This shifts the bottleneck from projector design to trace quality, dataset scale, and student context length.

8 Future Directions

Process and reward distillation. Reasoning traces and verifier signals can be distilled when direct use of a large teacher is too expensive at inference time.

Test-time compute distillation. A teacher may solve problems with sampling, search, best-of-N selection, or diffusion-style stitching; the student can then be trained to imitate the result in one forward pass [13].

Diffusion model distillation. Generative diffusion models often require many denoising steps. Distillation can compress the teacher's multi-step mapping into a student with fewer steps, sometimes one or two.

Multimodal distillation. Vision–language teachers can supervise single-modality students, but image patches and text tokens do not have an obvious one-to-one correspondence. Shared embedding spaces and caption-mediated supervision are common bridges.

Architecture-aware transfer. Future students may differ radically from their teachers: transformers, state-space models, convolutional networks, and mixture-of-experts systems encode different inductive biases. Orthogonal projection and other structure-preserving adapters are early examples of architecture-aware KD.

Takeaway. The lasting challenge is to decide which part of the teacher's knowledge is worth transferring and how to encode it as a stable training signal: a probability vector, feature map, relation graph, reasoning trace, reward, or denoising path.

References

[1] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *NIPS Workshop*, 2015. https://arxiv.org/abs/1503.02531.

[2] J. Gou et al. Knowledge distillation: A survey. *International Journal of Computer Vision*, 2021. https://arxiv.org/abs/2006.05525.

[3] T. Furlanello et al. Born again neural networks. *ICML*, 2018. https://arxiv.org/abs/1805.04770.

[4] A. Romero et al. FitNets: Hints for thin deep nets. *ICLR*, 2015. https://arxiv.org/abs/1412.6550.

[5] S. Zagoruyko and N. Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *ICLR*, 2017. https://arxiv.org/abs/1612.03928.

[6] W. Park et al. Relational knowledge distillation. *CVPR*, 2019. https://arxiv.org/abs/1904.05068.

[7] Y. Tian, D. Krishnan, and P. Isola. Contrastive representation distillation. *ICLR*, 2020. https://arxiv.org/abs/1910.10699.

[8] Z. Huang and N. Wang. Like what you like: Knowledge distill via neuron selectivity transfer. 2017. https://arxiv.org/abs/1707.01219.

[9] V. Sanh et al. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. 2019. https://arxiv.org/abs/1910.01108.

[10] W. Wang et al. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *NeurIPS*, 2020. https://arxiv.org/abs/2002.10957.

[11] X. Jiao et al. TinyBERT: Distilling BERT for natural language understanding. *EMNLP*, 2020. https://arxiv.org/abs/1909.10351.

[12] Y. Kim and A. Rush. Sequence-level knowledge distillation. *EMNLP*, 2016. https://arxiv.org/abs/1606.07947.

[13] R. Miles et al. Test-time scaling with diffusion language models via reward-guided stitching. 2026. https://arxiv.org/abs/2602.22871.

[14] B. Zhao and K. Cui. Decoupled knowledge distillation. *CVPR*, 2022. https://arxiv.org/abs/2203.08679.

[15] R. Miles and K. Mikolajczyk. Understanding the role of the projector in knowledge distillation. *AAAI*, 2024. https://arxiv.org/abs/2303.11098.

[16] C.-Y. Hsieh et al. Distilling step-by-step! Outperforming larger language models with less training data and smaller model sizes. *Findings of ACL*, 2023. https://arxiv.org/abs/2305.02301.

[17] P. Wang et al. SCOTT: Self-consistent chain-of-thought distillation. *ACL*, 2023.

[18] Y. Leviathan, M. Kalman, and Y. Matias. Fast inference from transformers via speculative decoding. *ICML*, 2023. https://arxiv.org/abs/2211.17192.

[19] R. Miles, I. Elezi, and J. Deng. VkD: Improving knowledge distillation using orthogonal projections. *CVPR*, 2024. https://arxiv.org/abs/2403.06213.

[20] H. Touvron et al. Training data-efficient image transformers and distillation through attention. *ICML*, 2021. https://arxiv.org/abs/2012.12877.

[21] DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. 2025. https://arxiv.org/abs/2501.12948.

[22] D. Auty, R. Miles, B. Kolbeinsson, and K. Mikolajczyk. Learning to project for cross-task knowledge distillation. *BMVC*, 2024. https://arxiv.org/abs/2403.14494.

[23] R. Miles, M. K. Yucel, B. Manganelli, and A. Saa-Garriga. MobileVOS: Real-time video object segmentation—contrastive learning meets knowledge distillation. *CVPR*, 2023. https://arxiv.org/abs/2303.07815.

[24] R. Miles, A. Lopez Rodriguez, and K. Mikolajczyk. Information theoretic representation distillation. *BMVC*, 2022. https://arxiv.org/abs/2112.00459.

[25] R. Miles and K. Mikolajczyk. Cascaded channel pruning using hierarchical self-distillation. *BMVC*, 2020. https://arxiv.org/abs/2008.06814.

[26] R. Miles, P. Reddy, I. Elezi, and J. Deng. VeLoRA: Memory efficient training using rank-1 sub-token projections. *NeurIPS*, 2024. https://arxiv.org/abs/2405.17991.

[27] R. Miles and K. Mikolajczyk. Reconstructing pruned filters using cheap spatial transformations. *ICCVW*, 2023. https://arxiv.org/abs/2110.12844.

[28] P. Chen et al. Distilling knowledge via knowledge review. *CVPR*, 2021. https://arxiv.org/abs/2104.09044.

[29] C. Xu et al. Knowledge distillation meets self-supervision. *ECCV*, 2020. https://arxiv.org/abs/2006.07114.

[30] B. Heo et al. A comprehensive overhaul of feature distillation. *ICCV*, 2019. https://arxiv.org/abs/1904.01866.

[31] N. Passalis and A. Tefas. Learning deep representations with probabilistic knowledge transfer. *ECCV*, 2018. https://arxiv.org/abs/1803.10837.