ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Energy-Based Dropout with Patch-Level Regularization

Tom Devynck[1], Bilal Faye[2], Djamel Bouchaffra[1], Nadjib Lazaar[3], Hanane Azzag[2], Mustapha Lebbah[1]

1 - DAVID Lab, UVSQ, Paris-Saclay University, UVSQ, Versailles, France
2- LIPN, UMR CNRS 7030 Sorbonne Paris Nord University, Villetaneuse, France
3- LISN, Paris-Saclay University, Saclay, France

Abstract.
Dropout is a widely used stochastic regularization technique, yet it overlooks structural dependencies within feature maps. We introduce PB-EDropout, an energy-based approach that preserves low-energy spatial patches within each channel while suppressing the rest. During training, candidate masks are sampled from Gibbs distributions and refined using genetic operators, and a running exponential moving average yields deterministic masks for inference. Experiments on shallow CNNs demonstrate that PB-EDropout consistently improves test accuracy over standard dropout, remains effective even with frozen masks, generates interpretable visualizations of discriminative features and are available here `https://github.com/Tom-Dvk/PB-EDropout/tree/main`.

1 Introduction

Deep learning architectures achieve remarkable performance across tasks but often suffer from redundancy and overfitting. Traditional dropout [1] mitigates overfitting by randomly omitting neurons, weights, or activations, reducing co-adaptation and improving generalization [2]. Beyond this, dropout has connections to ensemble learning and Bayesian inference [2], and structured variants such as Spatial Dropout [3] and Stochastic Depth [4] introduce randomness at the feature map or layer level. Adaptive schemes like Variational and Concrete Dropout [5] enable the network to learn data-dependent dropout probabilities. These ideas extend to Vision Transformers, where selectively dropping redundant visual tokens improves efficiency and generalization [6]. In CNNs, structured, data-driven methods such as DropCluster [7] enforce adaptive regularization by dropping correlated activations. Energy-based Dropout (EDropout) [8] further reframes dropout and pruning as an energy minimization problem, optimizing low-energy subnetworks.

Contributions. Building on the NeuroGame framework [9], which models neuronal cooperation via game theory and statistical physics, we introduce PB-EDropout, a topology-aware regularization method that bridges stochastic dropout with energy-based selection. Unlike EDropout [8], which computes energy from individual weights, PB-EDropout defines localized energy fields over neighboring neurons or patches, capturing spatially correlated activations and

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

emphasizing local structure. Additionally, unlike conventional dropout that only deactivates neurons during training, PB-EDropout preserves the pruning effect at inference, ensuring consistency while reducing the active network size.

2 Method

Algorithm 1: Energy-based patch dropout with population evolution

Input: batch $x \in \mathbb{R}^{N\times C\times H\times W}$, patch size k, keep rate p, population size S, mutation rate ρ, feedback rate η. Init G_t

foreach *channel* $c \in 1..C$ **do**
- Extract patches: $X_{i,p} \leftarrow \mathcal{P}(x_c)$ with kernel and stride k
- Compute energies $E_{i,p}$ and α_t
- Convert to Gibbs scores $\Pr(X_{i,p} \mid T_t)$
- **for** *candidate* $s \in 1..S$ **do**
  - Add small noise from a Gaussian $\mathcal{N}(\mu, \sigma^2)$ to Pr, select top-K patches, form $m^{(s)} \in \{0,1\}^{N\times L}$
  - Compute $\mathcal{E}\big(m^{(s)}\big)$
- $m_b \leftarrow \arg\min_s \mathcal{E}\big(m^{(s)}\big)$
- **foreach** *mask* $m^{(s)}$ **do**
  - Flip bits with probability ρ to obtain m'
  - Set $\hat{m}^{(s)}_{i,p} = \mathbf{1}\big[(1-\rho)m'_{i,p} + \rho m_{b,i,p} \geq 0.5\big]$
- Update global proxy G_t and α_t.
- Update EMA of best mask for channel c from m_b
- Apply m_b to reconstruct masked activations: $\tilde{x}_c = \mathcal{F}\big(m_b \odot \mathcal{P}(x_c)\big)$

Concatenate channels: $\tilde{x} = \mathrm{concat}_c\, \tilde{x}_c$.

Output: masked feature maps $\tilde{x}$.

We reinterpret dropout as a *structured, energy-based selection* acting on spatial patches of neurons rather than on individual units. Given a batch of convolutional feature maps $x \in \mathbb{R}^{N\times C\times H\times W}$, where N, C, H and W denote batch size, channels, height and width, respectively, our module processes each channel independently and applies a patch-wise masking layer prior to max-pooling. This section first introduces the NeuroGame-inspired energy model and local selection, followed by the population-based evolution mechanism. The procedure follows consists of six steps:

1- Patch extraction. For each channel $c \in 1..C$, we extract non-overlapping patches X of size $k \times k$ with stride k. Let $\mathcal{P}$ denote the patch extraction operator applied to feature map x_c: $\mathcal{P}(x_c) = \big[X_{c,1}, \ldots, X_{c,L}\big] \in \mathbb{R}^{L\times k\times k}$, $L = \frac{H}{k}\frac{W}{k}$, where L is the number of patches per channel [1]. After selection, the retained patches are folded back into the original spatial grid, producing a masked feature map of size $H \times W$.

2- Patch energy (Ising-inspired local consistency). For each channel, a feature map is divided into patches $X \in \mathbb{R}^{k\times k}$. We assign an Ising-inspired

[1] In all experiments, H and W are multiples of k. For example, on MNIST, feature maps of size 24×24 with $k = 2$ yield $L = 12 \times 12$ patches.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

energy:

$$E(X;\alpha_t,\beta) = \alpha_t \sum_{p\in X} \frac{1}{x_p} \;+\; \beta \sum_{\langle p,q\rangle \subset X} \frac{1}{x_p x_q}, \tag{1}$$

where p indexes spatial locations and $\langle p,q\rangle$ denotes all 8-neighbor pairs inside the patch. Large activations and coherent products $x_p x_q$ yield low energy, so low-energy patches correspond to strong, locally consistent responses. For each channel, this produces a per-sample, per-patch energy matrix $E \in \mathbb{R}^{N\times L}$. The single-site weight α_t is adapted during training. Let G_t be an exponential moving average (EMA) of a loss proxy (defined later); with feedback rate $\eta > 0$ we set $\alpha_t = 0.1\big(1 + \eta\, G_t\big)$. When patches are harder to remove (larger proxy), G_t increases, raising α_t and tightening the selection pressure.

3- Gibbs probabilities and temperature schedule. For each channel, we obtain an energy matrix $E \in \mathbb{R}^{N\times L}$, where $E_{i,p}$ is the energy of patch p for sample i. To enable stochastic exploration, these per-sample patch energies are converted into Gibbs probabilities with a temperature T_t that decreases during training:

$$\Pr(X_{i,p} \mid T_t) = \frac{\exp(-E_{i,p}/T_t)}{\sum_{j=1}^{L} \exp(-E_{i,j}/T_t)}, \qquad T_t = \frac{c}{\ln(1+t)}, \tag{2}$$

Initial population (top-p% sampling). A population of S candidate masks is generated from the Gibbs probabilities. For each candidate $s \in 1..S$, small noise is added to the probabilities, and the top-p% lowest-energy patches are retained, producing binary masks: $m_{i,p}^{(s)} = \begin{cases} 1, & \text{if patch } p \text{ belongs to the noisy top-}p\%, \\ 0, & \text{otherwise,} \end{cases}$

4- Population evolution: mutation, crossover, and selection. Energy-based ranking alone can be unstable, as small variations may shift which patches are selected. To stabilize selection, we refine the S noisy candidates through a lightweight evolutionary step operating in the discrete mask space $\{0,1\}^{N\times L}$. Each mask m is scored using the proxy

$$\mathcal{E}(m) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{L}\sum_{p=1}^{L} \|X_{i,p} - m_{i,p}X_{i,p}\|_1,$$

which measures the amount of activation mass removed. The best candidate over $s \in 1..S$ is $m_b = \arg\min_s \mathcal{E}(m^{(s)})$.

Mutation and crossover. Masks are perturbed by random bit flips with rate ρ, and the mutated mask m' is blended with the best mask to produce the next generation: $\hat{m}_{i,p} = \mathbf{1}\big[(1-\rho)m'_{i,p} + \rho\, m_{b,i,p} \geq 0.5\big]$. The feedback variable is updated via an EMA of the population's average proxy, $G_{t+1} = (1-\eta)G_t + \eta\,\overline{\mathcal{E}}$, where $\overline{\mathcal{E}} = \frac{1}{S}\sum_{s=1}^{S} \mathcal{E}(m^{(s)})$, which in turn modulates the single-site weight α_t.

5- EMA and inference masks. For each channel, we maintain an EMA tracking how frequently each patch is selected by the best training mask. At

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Table 1: Comparison of models by best validation accuracy and loss. Mask% is the patch-masking ratio.

	MNIST			CIFAR-10			CIFAR-100		
Model	Acc ↑	Loss ↓	Mask%	Acc ↑	Loss ↓	Mask%	Acc ↑	Loss ↓	Mask%
Baseline / No Dropout	99.290	0.0227	0%	77.230	1.1954	0%	47.050	3.2010	0%
Classic Dropout	99.270	0.0240	0%	77.350	**0.7808**	0%	45.140	**2.3475**	0%
PB-EDropout-Mask (ours)	99.380	0.0189	**15.28**	77.530	1.1944	**15.31**	47.240	3.1711	**15.31**
PB-EDropout-Full (ours)	**99.450**	**0.0189**	0%	**77.620**	1.2349	0%	**47.450**	3.1187	0%

inference, this EMA is binarized by retaining the top-p% entries, yielding a fixed deterministic mask (**MASK** mode). Disabling this mask restores the full, unpruned feature map (**FULL** mode).

6- Training and inference modes. During training, the best mask m_b is applied patchwise, and gradients flow only through the retained patches. At inference, we either apply the frozen EMA-based mask (MASK) or keep all activations intact (FULL).

The complete procedure is summarized in Algorithm 1, which reflects the implementation's control flow helper methods.

3 Experiments

We evaluate the proposed PB-EDropout mechanism on image classification using a shallow CNN composed of five convolutional layers. Two variants are examined: PB-EDropout-Full, which applies stochastic patch masking during training but uses no masking at inference, and PB-EDropout-Mask, which derives a deterministic inference mask by binarizing the exponential moving average of the best training masks. We compare these variants against a baseline CNN without regularization (No Dropout) and a model using standard random dropout (Classic Dropout). For all dropout-based methods, dropout is applied exclusively to the first convolutional layer. Experiments are conducted on MNIST, CIFAR-10, and CIFAR-100 [10], covering datasets of increasing complexity and resolution. The drop rate is fixed at 15%, meaning that at each forward pass, 15% of the spatial patches in the layer are masked. Regarding the initialization of the PB-EDropout hyperparameters, we set the population size to $S = 4$ and the mutation rate to $\rho = 0.1$. The global energy proxy G_0 is initialized to 1.0, and the feedback rate is set to $\eta = 0.05$. For the deterministic inference mask in PB-EDropout-Mask, the exponential moving average decay is set to 0.9.

Overall, the results in Table 1 show that PB-EDropout consistently improves generalization across datasets of increasing difficulty. On MNIST, both PB-EDropout variants outperform the Baseline and Classic Dropout, achieving the highest accuracy and lowest loss, demonstrating that patch-based regularization does not hinder performance on simple tasks and can stabilize training from early epochs. On CIFAR-10 and CIFAR-100, where feature maps exhibit

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

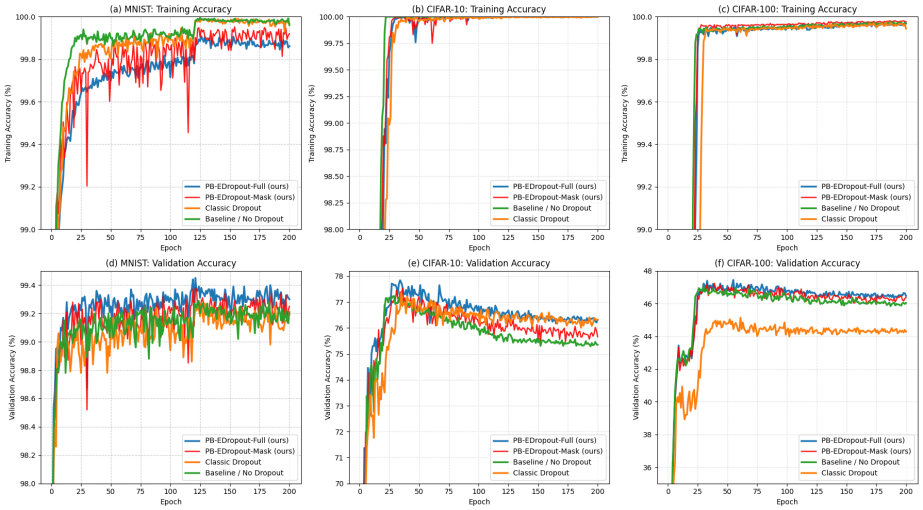

Fig. 1: Training and validation accuracy for a 5-layer CNN using PB-EDropout-Full, PB-EDropout-Mask, Classic Dropout, and Baseline.

stronger local correlations, PB-EDropout yields clearer performance gains: PB-EDropout-Full achieves the best accuracy on both datasets, while PB-EDropout-Mask attains the lowest test loss on CIFAR-10. These results indicate that population-based patch selection enhances the retention of informative features and that the two inference modes provide a controllable trade-off between probabilistic exploration (Full) and deterministic sparsity (Mask). Compared to Classic Dropout, which applies unstructured random omission, PB-EDropout selects spatially coherent patches based on the consistency of local activations. In our setup, approximately 15% of the patches in the dropout layer are masked, and the corresponding neurons remain unused during inference, a behavior not shared by standard dropout methods, which rely on the full network at test time. Even when numerical improvements are moderate, the learned masks are stable, human-interpretable, and aligned with visually discriminative regions of the input. This interpretability property distinguishes PB-EDropout from traditional dropout mechanisms, which cannot reveal the structural relevance of retained activations. Figure 1 strengthens these findings: although all models fit the training data comparably, PB-EDropout-Full sustains a consistently higher validation accuracy across epochs. This behavior indicates reduced overfitting and confirms that energy-based patch selection yields a more targeted and structurally meaningful regularization than random neuron suppression. Overall, PB-EDropout enhances generalization while providing interpretable spatial support maps that expose the subsets of features most consistently retained by the network, introducing a principled spatial sparsity that reveals which regions of the feature maps can be safely discarded without harming predictive performance.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

4 Conclusion

We introduced PB-EDropout, a topology-aware regularization method that selects low-energy spatial patches instead of dropping units at random. This patch-level strategy preserves coherent features, suppresses redundancy, and provides two complementary inference modes: PB-EDropout-Full for strong stochastic regularization and PB-EDropout-Mask for deterministic sparsity with interpretable activation maps. Across the experiment PB-EDropout consistently matches or exceeds classical dropout, yielding improved validation performance and robustness. These findings show that structured patch selection is an effective alternative to standard dropout in convolutional networks.

5 Acknowledgements

This work has received funding from the European Union Horizon Europe research and innovation programme under the Marie Sklodowska-Curie Actions (MSCA) grant agreement No. 101236749

References

[1] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580, 2012.

[2] Alex Labach, Hojjat Salehinejad, and Shahrokh Valaee. Survey of dropout methods for deep neural networks. *CoRR*, abs/1904.13310, 2019.

[3] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 648–656. IEEE Computer Society, 2015.

[4] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q. Weinberger. Deep networks with stochastic depth. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, volume 9908 of *Lecture Notes in Computer Science*, pages 646–661. Springer, 2016.

[5] Yarin Gal, Jiri Hron, and Alex Kendall. Concrete dropout. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 3581–3590, 2017.

[6] Chengcheng Gao, Xu Zhang, Shu Li, et al. Patchdropout: Economizing vision transformers via patch-level sampling. *arXiv preprint arXiv:2208.07220*, 2022.

[7] Liyan Chen, Philip Gautier, and Sergül Aydöre. Dropcluster: A structured dropout for convolutional networks. *CoRR*, abs/2002.02997, 2020.

[8] Hojjat Salehinejad and Shahrokh Valaee. Edropout: Energy-based dropout and pruning of deep neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, 33(10):5279–5292, 2022.

[9] B. Faye M. Lebbah D. Bouchaffra, F. Ykhlef and H. Azzag. Game theory meets statistical physics: A novel deep neural networks design. *IEEE Transactions on Cybernetics*, 191, 2025.

[10] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 and cifar-100 datasets. Technical report, University of Toronto, 2009. https://www.cs.toronto.edu/ kriz/cifar.html.