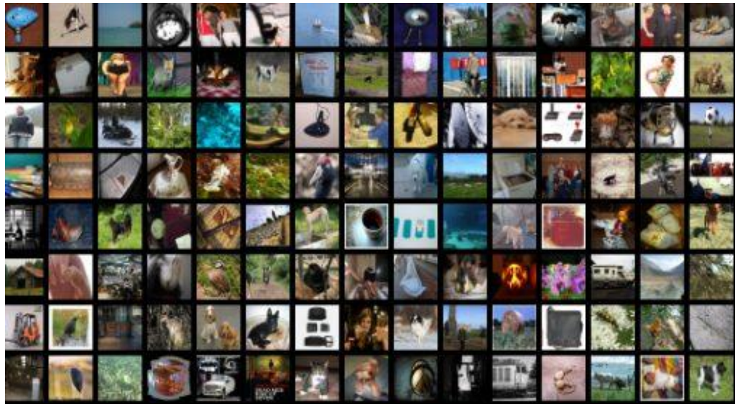
Invex Optimization: Theory and Applications for Signal/Image Processing and Machine Learning

Samuel Pinilla

Diamond Light Source, Rutherford Appleton Laboratory, UK

Problem formulation landscape

Layer 1 (Data)

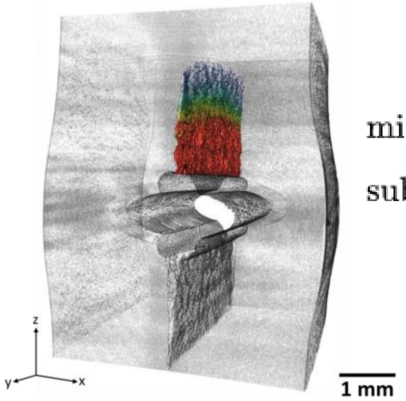


Data chosen in terms of

- Noise Level
- Colour properties
- Depth information
- Resolution
- Quality
- Dataset size
- Fourier frequencies

⇄

Layer 2 (Mathematics)



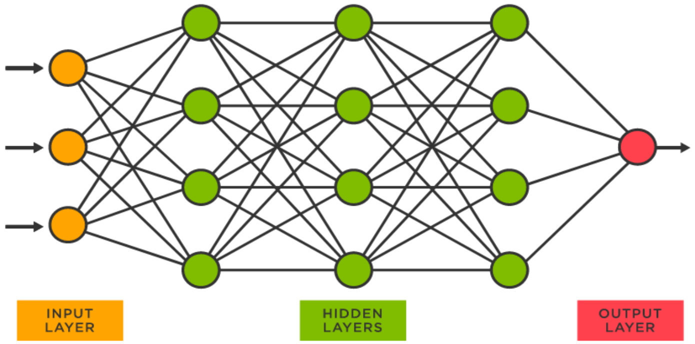
$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \\ & h_j(x) = 0, \end{aligned}$$

Model chosen in terms of

- Image formation process
- Optics resolution
- Sensor model (noise)
- Convex/Non-convex optimisation
- Light source
- Fourier optics theory

⇄

Layer 3 (Software)

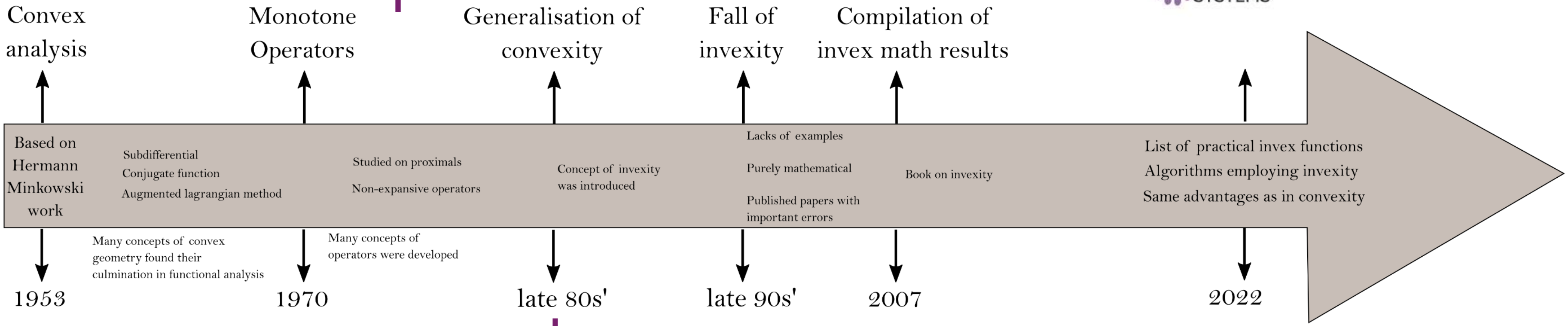


Software chosen in terms of

- Hardware capabilities
- Performance
- Reproducibility
- Costs
- Flexibility
- Distributed programming
- Scalability

Timeline in Optimization

$$\operatorname{Prox}_g(\boldsymbol{u}) = \underset{\boldsymbol{x}\in\mathbb{R}^n}{\arg\min}\left(g(\boldsymbol{x}) + \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{u}\|_2^2\right)$$



Definition 2 (Invexity). Let $f:\mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz; then f is invex if there exists a function $\eta:\mathbb{R}^n\times\mathbb{R}^n\to\mathbb{R}^n$ such that

$$f(\boldsymbol{x}) - f(\boldsymbol{y}) \geq \zeta^T\eta(\boldsymbol{x},\boldsymbol{y}),$$

$\forall \boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^n, \forall\zeta\in\partial f(\boldsymbol{y})$.

Pinilla, S., Mu, T., Bourne, N., & Thiyagalingam, J. (2022). Improved imaging by invex regularizers with global optima guarantees. Advances in Neural Information Processing Systems, 35, 10780-10794.

Invexity Concept

Definition *Assume $X \subseteq R^n$ is an open set. The differentiable function $f : X \to R$ is invex if there exists a vector function $\eta : X \times X \to R^n$ such that*

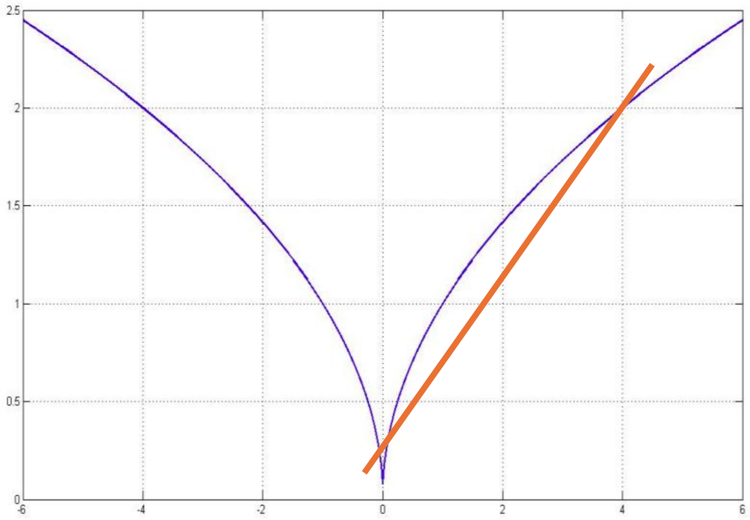
$$f(x) - f(y) \geq \eta(x, y)^T \nabla f(y), \quad \forall x, y \in X.$$

It is well known that a convex function simply satisfies this definition for $\eta(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x} - \boldsymbol{y}$.

Key property

Theorem *Let $f : X \to R$ be differentiable. Then f is invex if and only if every stationary point is a global minimizer.*

Invexity vs Convexity



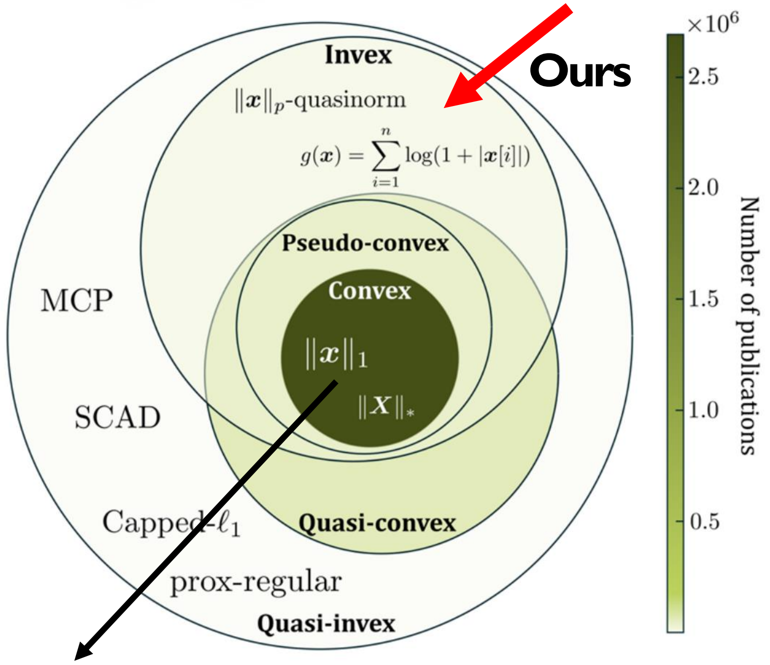
While convexity is not sufficient to characterize the above function invex.

Pinilla, S., Mu, T., Bourne, N., & Thiyagalingam, J. (2022). Improved imaging by invex regularizers with global optima guarantees. Advances in Neural Information Processing Systems, 35, 10780-10794.

Why Invexity?

Assumption	Stable recovery	Best performance	Efficient recovery algorithm
Convexity	✗	✗	✓
Invexity	✓	✓	✓

Universe of optimisable functions



Enjoys extensive maths, but it does not provide the best performance; limited options (smallest set)

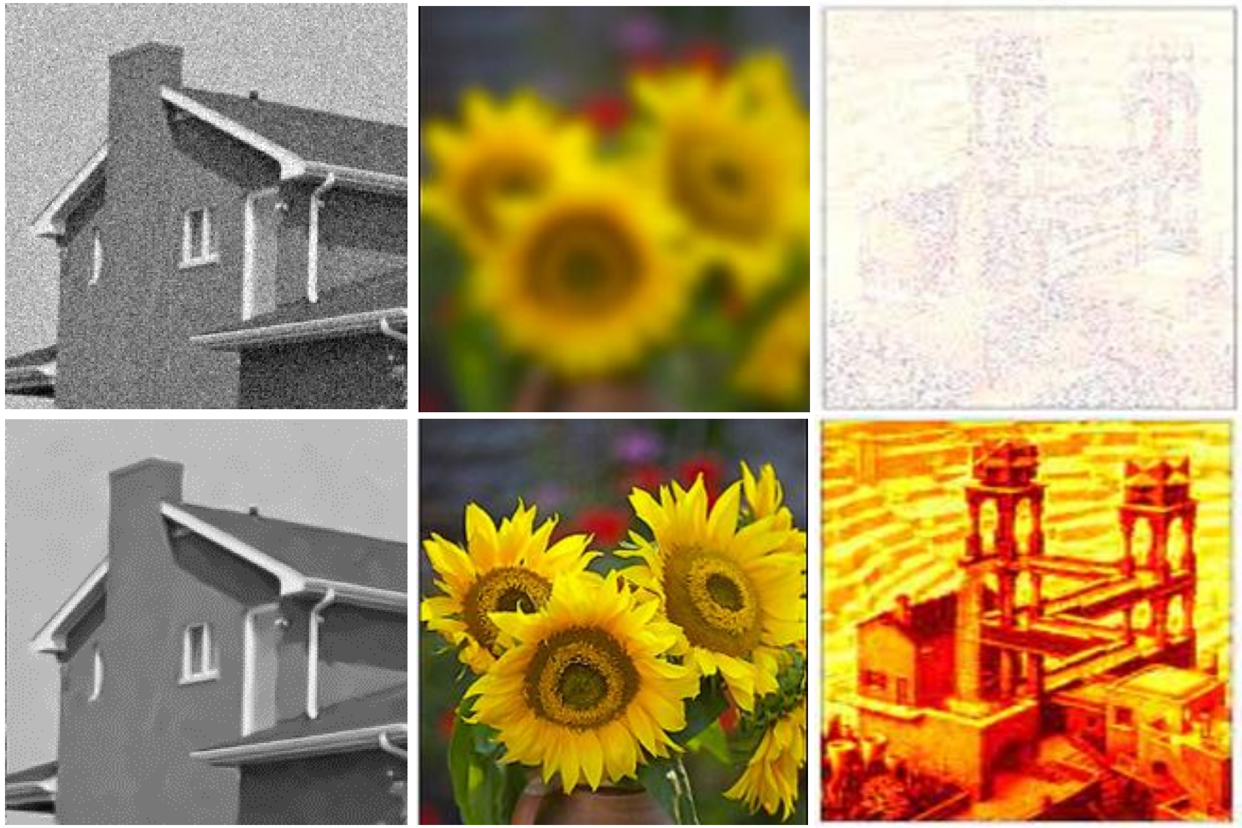
Our contributions

- Wider range of regularizer for imaging
- We provide **uniqueness recovery**
- Efficient algorithms
- Best imaging performance

Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

Example

Denoising | Deblurring | Compressive sensing



Formulation

$$\underset{x \in \mathbb{R}^n}{\text{Minimize}} \quad F(\boldsymbol{x}) = \boxed{f(\boldsymbol{x})} + \boxed{g(\boldsymbol{x})}$$

Fidelity term: $f(\boldsymbol{x})$

Regularizer: $g(\boldsymbol{x})$

Challenges

- Selection of the regularizer (e.g. convex, Invex)
- Successful Recovery depends on the regularizer
- Preserve image quality
- Development of efficient algorithms

Wen, F., Chu, L., Liu, P., & Qiu, R. C. (2018). A survey on nonconvex regularization-based sparse and low-rank recovery in signal processing, statistics, and machine learning. IEEE Access, 6, 69883-69906.

Examples: Invex Regularizers

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} (|\boldsymbol{x}[i]| + \epsilon)^p \text{ , for } p \in (0,1) \text{ and } \epsilon \geq (p(1-p))^{\frac{1}{2-p}} \text{ ,}$$

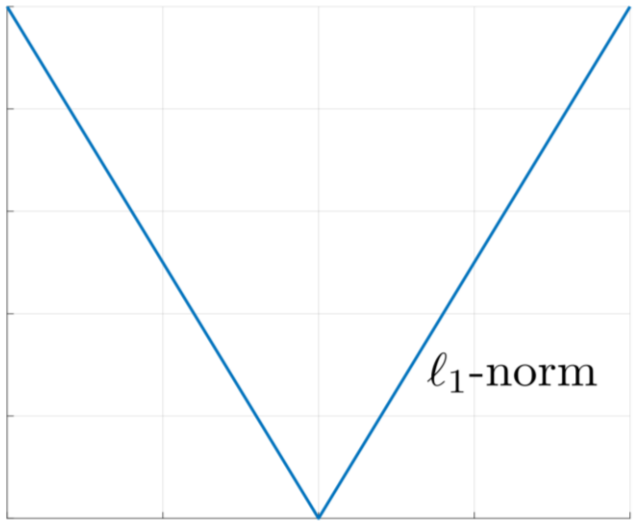
$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \log(1 + |\boldsymbol{x}[i]|),$$

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{|\boldsymbol{x}[i]|}{2 + 2|\boldsymbol{x}[i]|},$$

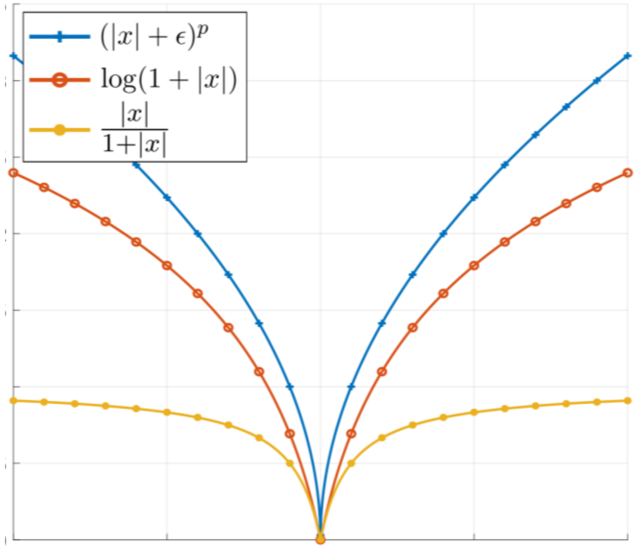
$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{\boldsymbol{x}^2[i]}{1 + \boldsymbol{x}^2[i]},$$

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \log(1 + |\boldsymbol{x}[i]|) - \frac{|\boldsymbol{x}[i]|}{2 + 2|\boldsymbol{x}[i]|}.$$

Convex



Invex



Pinilla, S., Mu, T., Bourne, N., & Thiyagalingam, J. (2022). Improved imaging by invex regularizers with global optima guarantees. Advances in Neural Information Processing Systems, 35, 10780-10794.

Examples: Invex Fidelity Losses

$$\text{(Cauchy)} \quad f(\boldsymbol{x}) = \sum_{i=1}^{m} \log\left(1 + \frac{\boldsymbol{x}^2[i]}{\delta^2}\right)$$

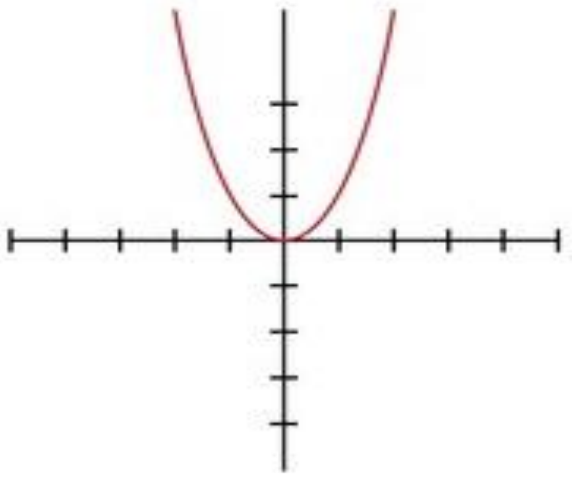
$$\text{(Geman-McClure)} \quad f(\boldsymbol{x}) = \sum_{i=1}^{m} \frac{2\boldsymbol{x}^2[i]}{\boldsymbol{x}^2[i] + 4\delta^2}$$

$$\text{(Welsh)} \quad f(\boldsymbol{x}) = \sum_{i=1}^{n} 1 - \exp\left(\frac{-\boldsymbol{x}^2[i]}{2\delta^2}\right)$$

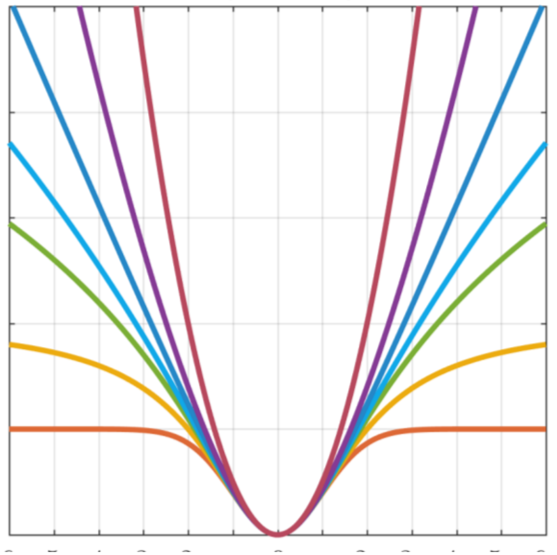
$$\text{(Adaptive robust)} \quad f(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{|\alpha - 2|}{\alpha}\left(\left(\frac{(\boldsymbol{x}[i]/c)^2}{|\alpha - 2|} + 1\right)^{\alpha/2} - 1\right)$$

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} \log\left(1 + \boldsymbol{x}^2[i]\right) - \frac{\boldsymbol{x}^2[i]}{2\boldsymbol{x}^2[i] + 2}$$

Convex



Adaptive Robust (Invex)



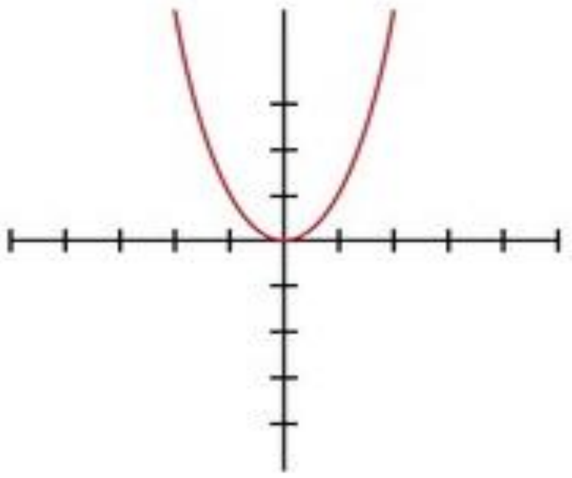
Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

Tools for Invexity

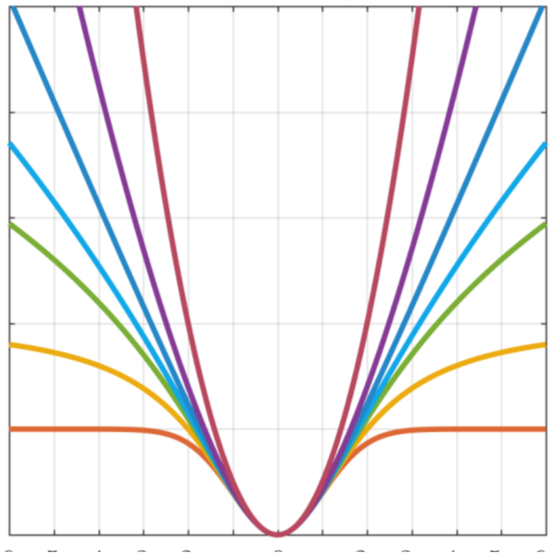
1. *every stationary point of f is a global minimizer*
2. *f does not have stationary points*
3. *for $n = 1$ if first derivative of f satisfies $\sigma' > 0$*

These organized mathematical tools are useful to construct invex functions, which we believe are missing in the literature.

Convex



Adaptive Robust (Invex)

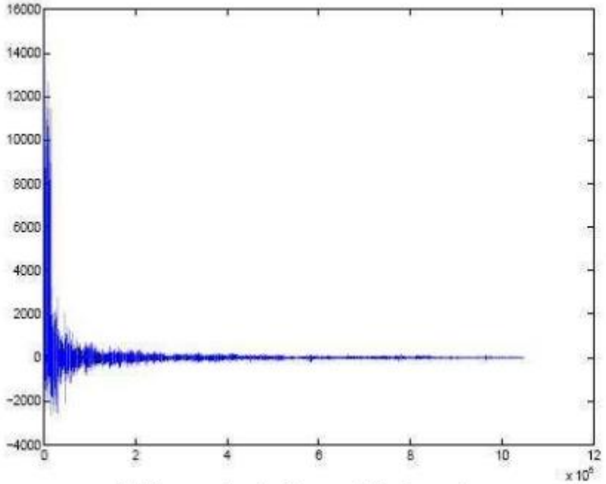


Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

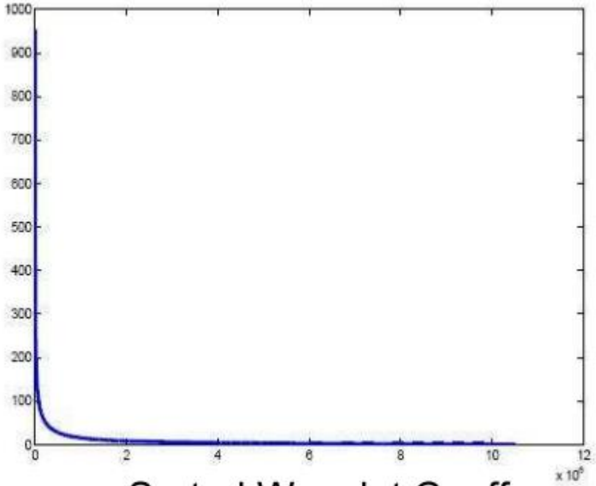
Sparse Signal Representation

- Wavelet coefficients of natural scenes exhibit the $(1/n)$-decay[†].

1 Megapixel Image



Wavelet Coefficients



Sorted Wavelet Coeff.

Candès, E. J., & Wakin, M. B. (2008). An introduction to compressive sampling. IEEE signal processing magazine, 25(2), 21-30.

Sparse Signal Representation

Basis Pursuit $\rightarrow$ find the sparsest approximation of $\mathbf{x}$

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \mathbf{x} = \boldsymbol{\Psi}\boldsymbol{\alpha}$$

where $\|\boldsymbol{\alpha}\|_1 = \sum_i |\alpha_i|$.

- BP decomposes a signal into a superposition of dictionary elements having the smallest ℓ_1-norm among all such decompositions.

Candès, E. J., & Wakin, M. B. (2008). An introduction to compressive sampling. IEEE signal processing magazine, 25(2), 21-30.

Recovery Guarantees: For Convex Regularizer

A matrix $\mathbf{A} \in R^{m \times n}$ satisfies the Restricted Isometry Property if there exists a constant $\delta > 0$ such that

$$(1-\delta)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1+\delta)\|\mathbf{x}\|_2^2$$

with high probability†.

Outline of the proof:

1. Show that for a fixed sparse vector $\mathbf{x}$, $\|\mathbf{A}\mathbf{x}\|_2^2 \approx \|\mathbf{x}\|_2^2$ with high probability.
2. Count up the "number" of sparse vectors, and show that $\|\mathbf{A}\mathbf{x}\|_2^2 \approx \|\mathbf{x}\|_2^2$ for all of them with high probability.

Key message:

Same recovery guarantees are valid for the presented invex regularizers.

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} (|\boldsymbol{x}[i]| + \epsilon)^p,$$

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \log(1 + |\boldsymbol{x}[i]|),$$

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{|\boldsymbol{x}[i]|}{2 + 2|\boldsymbol{x}[i]|},$$

$$g(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{\boldsymbol{x}^2[i]}{1 + \boldsymbol{x}^2[i]},$$

Candès, E. J., & Wakin, M. B. (2008). An introduction to compressive sampling. IEEE signal processing magazine, 25(2), 21-30.

Pinilla, S., Mu, T., Bourne, N., & Thiyagalingam, J. (2022). Improved imaging by invex regularizers with global optima guarantees. Advances in Neural Information Processing Systems, 35, 10780-10794.

Family of Invex Functions

Definition *(Family of Functions) Let $h : \mathbb{R}^n \to \mathbb{R}$ such that $h(\boldsymbol{x}) = \sum_{i=1}^{n} r(|\boldsymbol{x}[i]|)$, where $r : [0, \infty) \to [0, \infty)$ satisfies*

1. *$r(0) = 0$, and $r'(w) > 0$ for $w \in (0, \infty)$;*
2. *$r(w)/w^2$ is non-increasing on $(0, \infty)$.*

Properties

Theorem 8 (Algebraic Properties) *Let $f, g : \mathbb{R}^n \to \mathbb{R}$ be two functions as in Definition, such that $f(\boldsymbol{x}) = \sum_{i=1}^{n} s_f(|\boldsymbol{x}[i]|)$, and $g(\boldsymbol{x}) = \sum_{i=1}^{n} s_g(|\boldsymbol{x}[i]|)$. Then the following functions are invex:*

- *$f(\boldsymbol{x})$, and $g(\boldsymbol{x})$;*
- *$q(\boldsymbol{x}) = \alpha f(\boldsymbol{x}) + \beta g(\boldsymbol{x})$ for every $\alpha, \beta \geq 0$;*
- *$q(\boldsymbol{x}) = \sum_{i=1}^{n} (s_f \circ s_g)(|\boldsymbol{x}[i]|)$;*
- *$q(\boldsymbol{x}) = \sum_{i=1}^{n} (s_f \cdot s_g)(|\boldsymbol{x}[i]|)$;*
- *$q(\boldsymbol{x}) = f(\boldsymbol{x}) - g(\boldsymbol{x})$, if $s'_f(w) > s'_g(w)$ and $\frac{s_f(w) - s_g(w)}{w^{c+1}}$ non-increasing on $(0, \infty)$ for some integer $c \geq 0$;*
- *$q(\boldsymbol{x}) = \sum_{i=1}^{n} \min(s_f(|\boldsymbol{x}[i]|), s_g|\boldsymbol{x}[i]|)$;*
- *$q(\boldsymbol{x}) = \sum_{i=1}^{n} \max(s_f(|\boldsymbol{x}[i]|), s_g|\boldsymbol{x}[i]|)$.*

Key message:

- Family of invex functions closed under summation.
- It paves the way for establishing theoretical guarantees for global optima.
- Secondly, it bestows practical benefits to practitioners.
- Offer a systematic methodology for constructing invex functions.
- Convexity is not always preserved under minimum and product.

Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

Examples

Lemma (Invex Mappings) *The following functions for $p \in (0,1)$, $\epsilon > 0$, and $\boldsymbol{\omega} \in \mathbb{R}^n_{++}$ are invex:*

$$q(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{1}{a} \log(1 + a|\boldsymbol{x}[i]|), a \in (0,1],$$

$$q(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{|\boldsymbol{x}[i]|}{a + a|\boldsymbol{x}[i]|}, a \geq 2$$

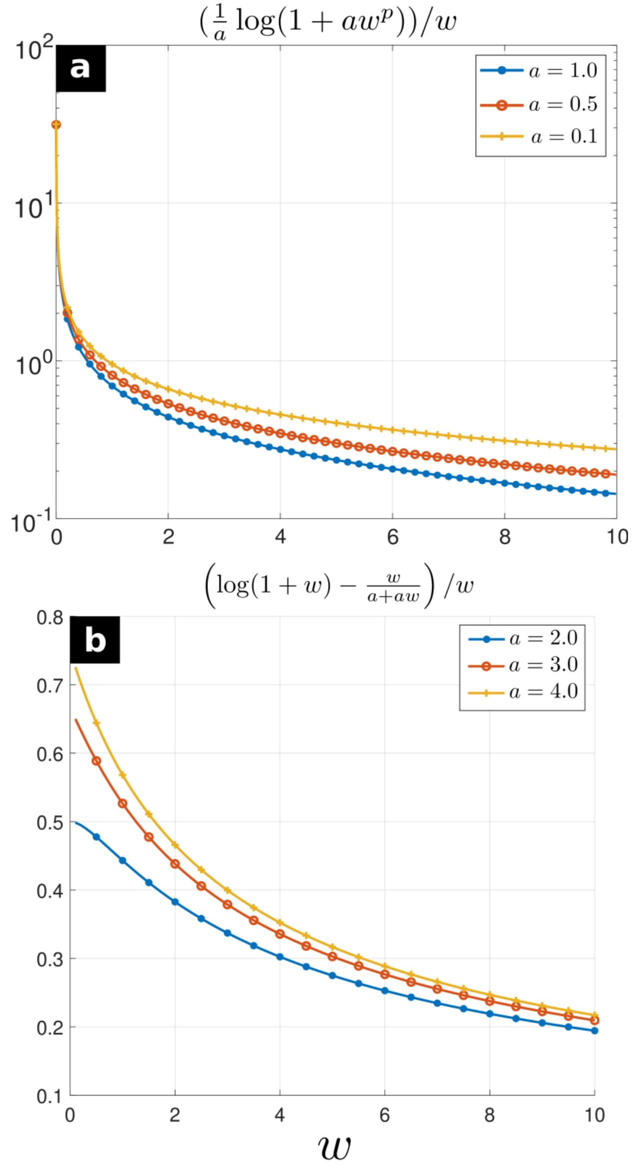
$$q(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{1}{a} \log(1 + a\left(|\boldsymbol{x}[i]| + \epsilon\right)^p), a \in (0,1],$$

$$q(\boldsymbol{X}) = \sum_{i=1}^{m} \left(\sum_{j=1}^{n} (|\boldsymbol{X}[i,j]| + \epsilon)^p \right)^{q/p}, q \geq p,$$

$$q(\boldsymbol{x}) = \sum_{i=1}^{n} \log\left(1 + |\boldsymbol{x}[i]|\right) - \frac{|\boldsymbol{x}[i]|}{a + a|\boldsymbol{x}[i]|}, a \geq 2,$$

$$q(\boldsymbol{x}) = \sum_{i=1}^{n} \boldsymbol{\omega}[i] \left(\frac{1}{a} \log(1 + a|\boldsymbol{x}[i]|) - \frac{|\boldsymbol{x}[i]|}{1 + a|\boldsymbol{x}[i]|/2} \right), a \in (0,1],$$

Verifying Definition



Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

Proximal Operator

- proximal operator of $f : \mathbf{R}^n \to \mathbf{R} \cup \{+\infty\}$ is

$$\mathbf{prox}_{\lambda f}(v) = \underset{x}{\operatorname{argmin}} \left(f(x) + (1/2\lambda)\|x - v\|_2^2 \right)$$

  with parameter $\lambda > 0$

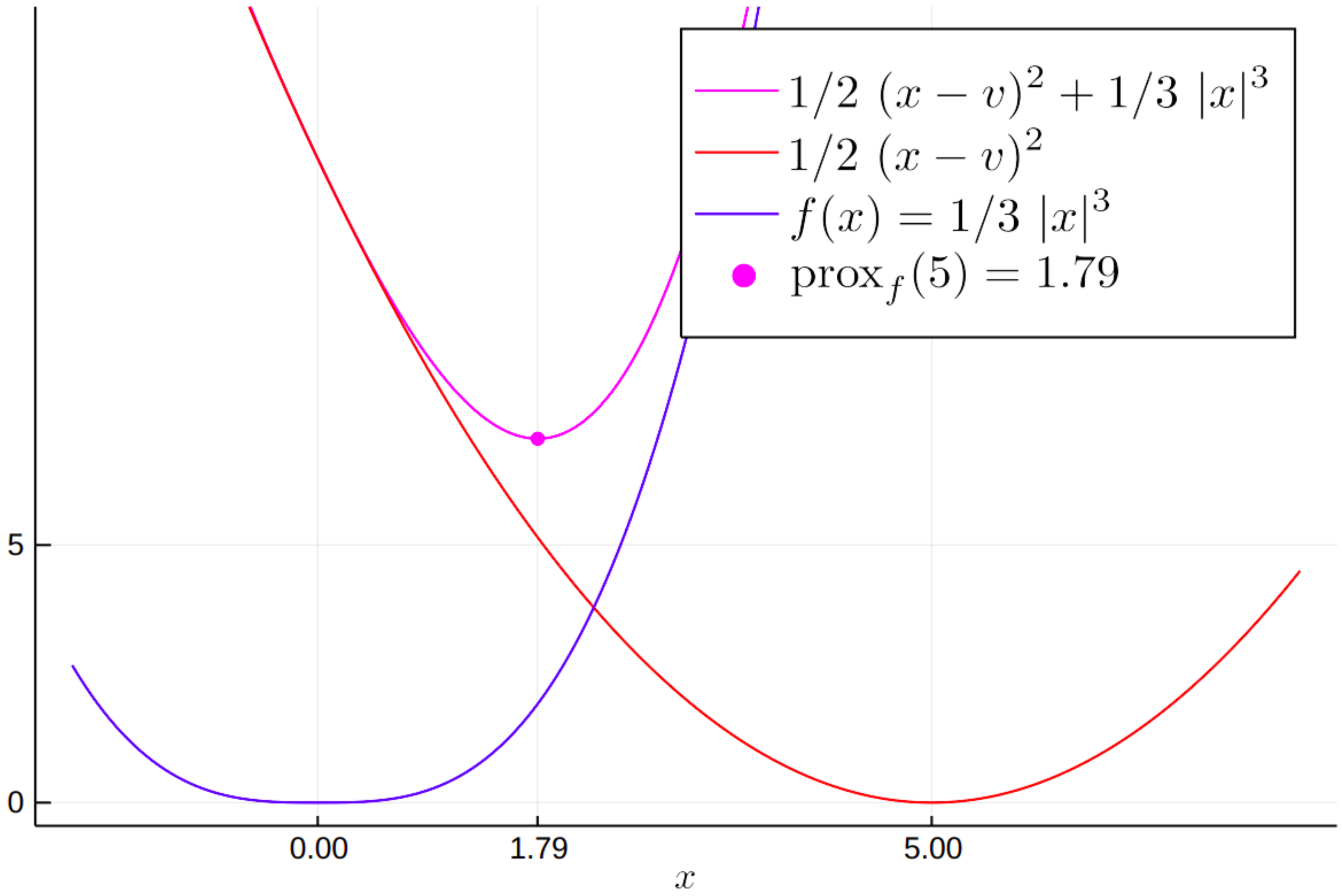
- f may be nonsmooth, have embedded constraints, ...

- evaluating $\mathbf{prox}_f$ involves solving a convex optimization problem

---

Parikh, N., & Boyd, S. (2014). Proximal algorithms. Foundations and trends® in Optimization, 1(3), 127-239.

Proximal Operator



Parikh, N., & Boyd, S. (2014). Proximal algorithms. Foundations and trends® in Optimization, 1(3), 127-239.

Proximal Operator: Family of Invex Function

$$\text{minimize} \quad f(x) + g(x)$$

- method:

$$x^{k+1} := \mathbf{prox}_{\lambda^k g}(x^k - \lambda^k \nabla f(x^k))$$

- converges with rate $O(1/k)$ when ∇f is Lipschitz continuous with constant L and step sizes are $\lambda^k = \lambda \in (0, 1/L]$

- $0 \in \nabla f(x^\star) + \partial g(x^\star)$ if and only if

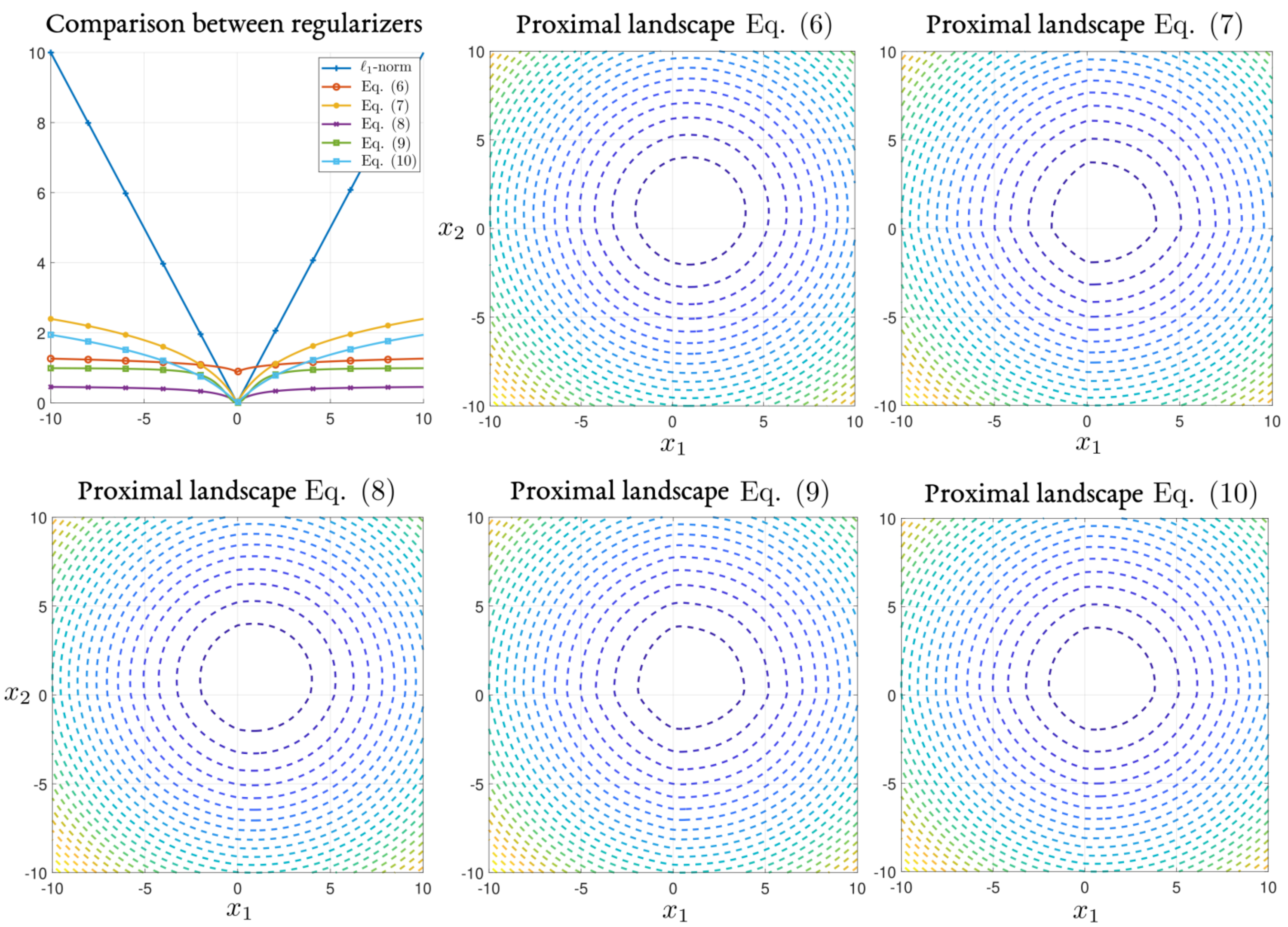
$$x^\star = (I + \lambda \partial g)^{-1}(I - \lambda \nabla f)(x^\star)$$

i.e., $x^\star$ is a fixed point of *forward-backward operator*

Consequence of using studied family of invex functions

Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

Proximal Operator: Family of Invex Function



(6) $$g(\boldsymbol{x}) = \sum_{i=1}^{n} (|\boldsymbol{x}[i]| + \epsilon)^p ,$$

(7) $$g(\boldsymbol{x}) = \sum_{i=1}^{n} \log(1 + |\boldsymbol{x}[i]|),$$

(8) $$g(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{|\boldsymbol{x}[i]|}{2 + 2|\boldsymbol{x}[i]|},$$

(9) $$g(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{\boldsymbol{x}^2[i]}{1 + \boldsymbol{x}^2[i]},$$

(10) $$g(\boldsymbol{x}) = \sum_{i=1}^{n} \log(1 + |\boldsymbol{x}[i]|) - \frac{|\boldsymbol{x}[i]|}{2 + 2|\boldsymbol{x}[i]|}.$$

Pinilla, S., Mu, T., Bourne, N., & Thiyagalingam, J. (2022). Improved imaging by invex regularizers with global optima guarantees. Advances in Neural Information Processing Systems, 35, 10780-10794.

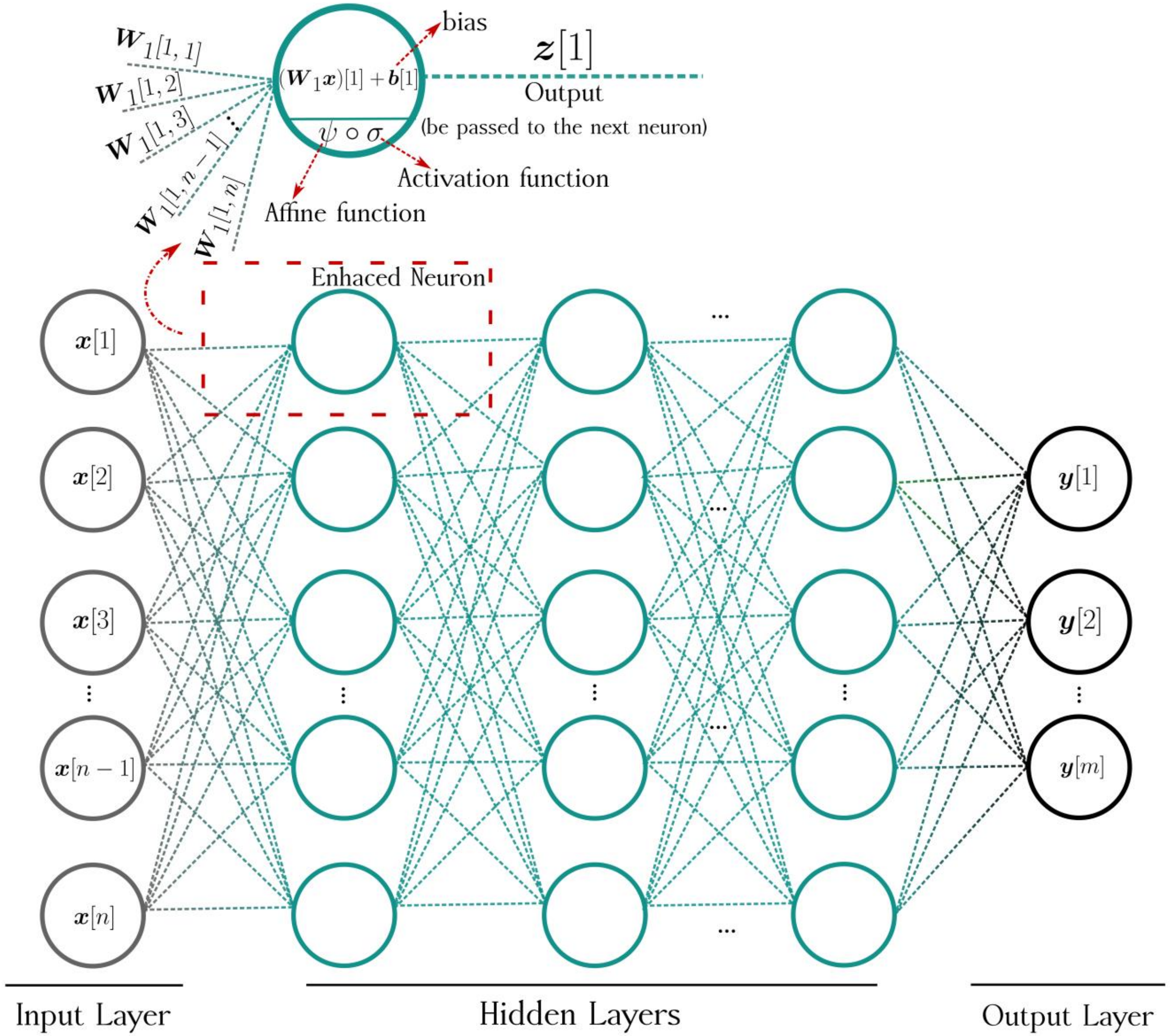
Speed of Proximal for Invex Functions

```
invex2D_kernel = cp.RawKernel(r'''
        extern "C"
        __global__ void invex2DFilter(const float *x, float *y, float lamb, float q, int size){

            int tid = blockDim.x * blockIdx.x + threadIdx.x;

            if (tid < size){
                float beta = powf(2.0*lamb*(1.0-q),1.0/(2.0-q));
                float tau  = beta + lamb*q*powf(beta,q-1.0);
                float sign = -2.0*signbit(x[tid]) + 1.0;

                y[tid] = 0.0f;

                if(fabs(x[tid]) > tau){
                    float z = beta + (fabs(x[tid])-beta)/2.0;

                    for(int k=0;k < 4;k++){
                        z = fabs(x[tid]) - lamb*q*powf(z,q-1.0);
                    }

                    y[tid] = sign * z;

                }
            }
        }
        ''', 'invex2DFilter')
```

	equation (7)	ℓ_p-quasinorm	equation (9)	ℓ_1-norm
Time	$2.8ms$	$1.06ms$	$0.86ms$	$0.66ms$

Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

Machine Learning



Elements

- Activation functions
- Weights.

Pinilla, S., Sanabria, A., Bi, J., & Egiazarian, K. (2025). What Makes Neural Networks Trainable? Invexity as a Structural Design Principle in AI.

Classification of Activation Functions

Name	Activation function	Range	Type
Identity (31)	$\sigma(x) = x$	$(-\infty, \infty)$	Convex
Absolute value (31)	$\sigma(x) = \lvert x \rvert$	$[0, \infty)$	
ReLU (32)	$\sigma(x) = \max\{0, x\}$	$[0, \infty)$	
Leaky ReLU (33)	$\sigma(x) = \begin{cases} x & \text{if } x > 0 \\ 0.01x & \text{otherwise} \end{cases}$	$(-\infty, \infty)$	
Softmax (14)	$\sigma_i(\boldsymbol{x}) = \frac{e^{\boldsymbol{x}[i]}}{\sum_{j=1}^{n} e^{\boldsymbol{x}[j]}}$	$(0, 1)$	
Softplus (34)	$\sigma(x) = \frac{1}{\beta}\log(1 + e^{\beta x}),\ \beta > 0$	$(0, \infty)$	
Generalized Sigmoid (35)	$\sigma(x) = \frac{\alpha}{1+e^{-\beta x}},\ \alpha, \beta > 0$	$(0, \alpha)$	Invex
Shifted-Scaled Sigmoid (36)	$\sigma(x) = \frac{1}{1+e^{-a(x-b)}},\ a, b > 0$	$(0, 1)$	
SiLU (37)	$\sigma(x) = \frac{x}{1+e^{-x}}$	$[\approx -0.27, 1)$	
SigLin (38)	$\sigma(x) = \frac{1}{1+e^{-x}} + 0.1x$	$(-\infty, \infty)$	
Soft-root-sign (39)	$\sigma(x) = \frac{x}{x/2+e^{-x/3}}$	$[\frac{6}{(3-2e)}, 2)$	
Elliot (40)	$\sigma(x) = \frac{0.5x}{1+\lvert x \rvert} + 0.5$	$(0, 1)$	
Log-ReLU (41)	$\sigma(x) = \log(\beta \max\{x, 0\} + 1),\ \beta > 0$	$[0, \infty)$	
Suish (23)	$\sigma(x) = \max\{x, xe^{-\lvert x \rvert}\}$	$[\approx -0.36, \infty)$	
SELU (42)	$\sigma(x) = \lambda \cdot (\max\{0, x\} + \min\{0, \alpha(\exp(x) - 1)\}), \lambda \approx 1.05, \alpha \approx 1.67$	$(\approx -1.67, \infty)$	
Biopolar Sigmoid (43)	$\sigma(x) = \frac{1-e^{-x}}{1+e^{-x}}$	$(-1, 1)$	
Hyperbolic tangent (44)	$\sigma(x) = \tanh(x)$	$(-1, 1)$	
TanhSig (23)	$\sigma(x) = (x + \tanh(x))/(1 + \exp(-x))$	$\approx [-0.47, \infty)$	
E-swish (45)	$\sigma(x) = \beta \frac{x}{1+e^{-x}},\ \beta > 0$	$[\approx -0.27\beta, \infty)$	
GELU (46)	$\sigma(x) = x\Phi(x)$ ie., $\Phi(x) = \frac{1}{2}(1 + \text{erf}(\frac{x}{\sqrt{2}}))$	$[\approx -0.16, \infty)$	

Tools

1. *every stationary point of f is a global minimizer*
2. *f does not have stationary points*
3. *for $n = 1$ if first derivative of f satisfies $\sigma' > 0$*

Example

Log-ReLU: This function is defined as $\sigma(x) = \log(\beta \max\{x, 0\} + 1)$ for $\beta > 0$. $\partial\sigma(x)$ is given as

$$\partial\sigma(x) = \begin{cases} 0 & \text{if } x < 0 \\ \frac{\beta}{\beta x + 1} & \text{if } x > 0 \\ [0, \beta] & \text{if } x = 0 \end{cases} .$$

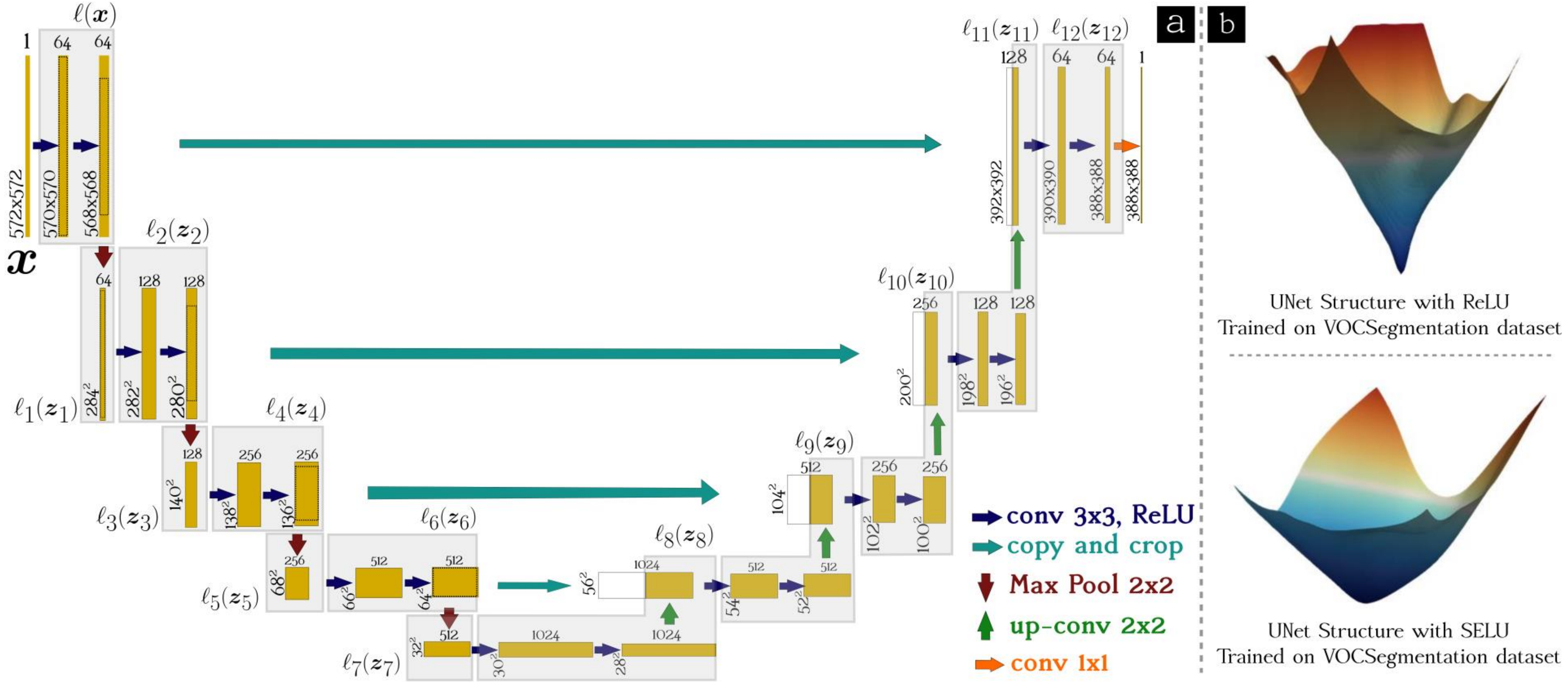
Pinilla, S., Sanabria, A., Bi, J., & Egiazarian, K. (2025). What Makes Neural Networks Trainable? Invexity as a Structural Design Principle in AI.

Invex Neural Networks

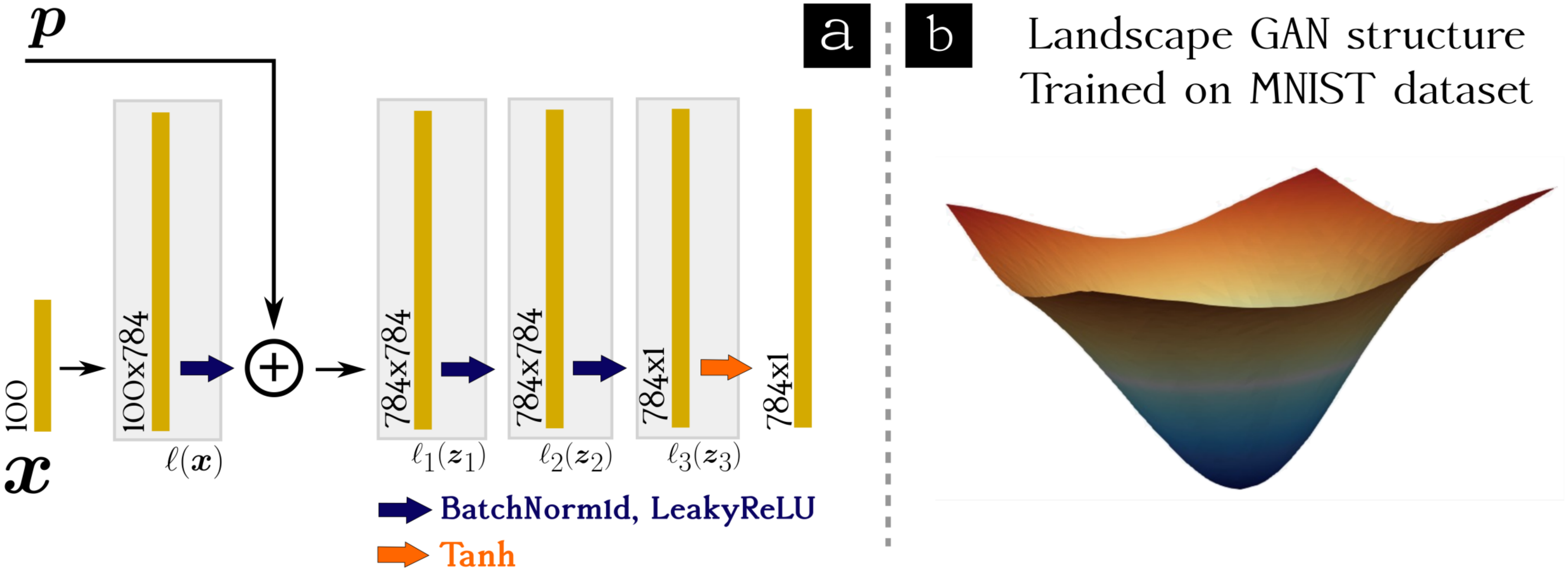
Mathematically, without loss of generality, an *enhanced perceptron layer* of a deep neural network $\mathcal{N}_\theta$ defined as $\ell : \mathbb{R}^r \times \mathbb{R}^d \to \mathbb{R}^d$ is given by

$$\ell(\boldsymbol{z}, \boldsymbol{p}) = \sigma\left(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}\right) + \boldsymbol{p},$$

Proposition 2. *If $\ell(\boldsymbol{z}, \boldsymbol{p})$ is an enhanced perceptron layer then is invex. Additionally, any feed-forward composition of an arbitrary number of enhanced hidden layers, yields to an invex network.*

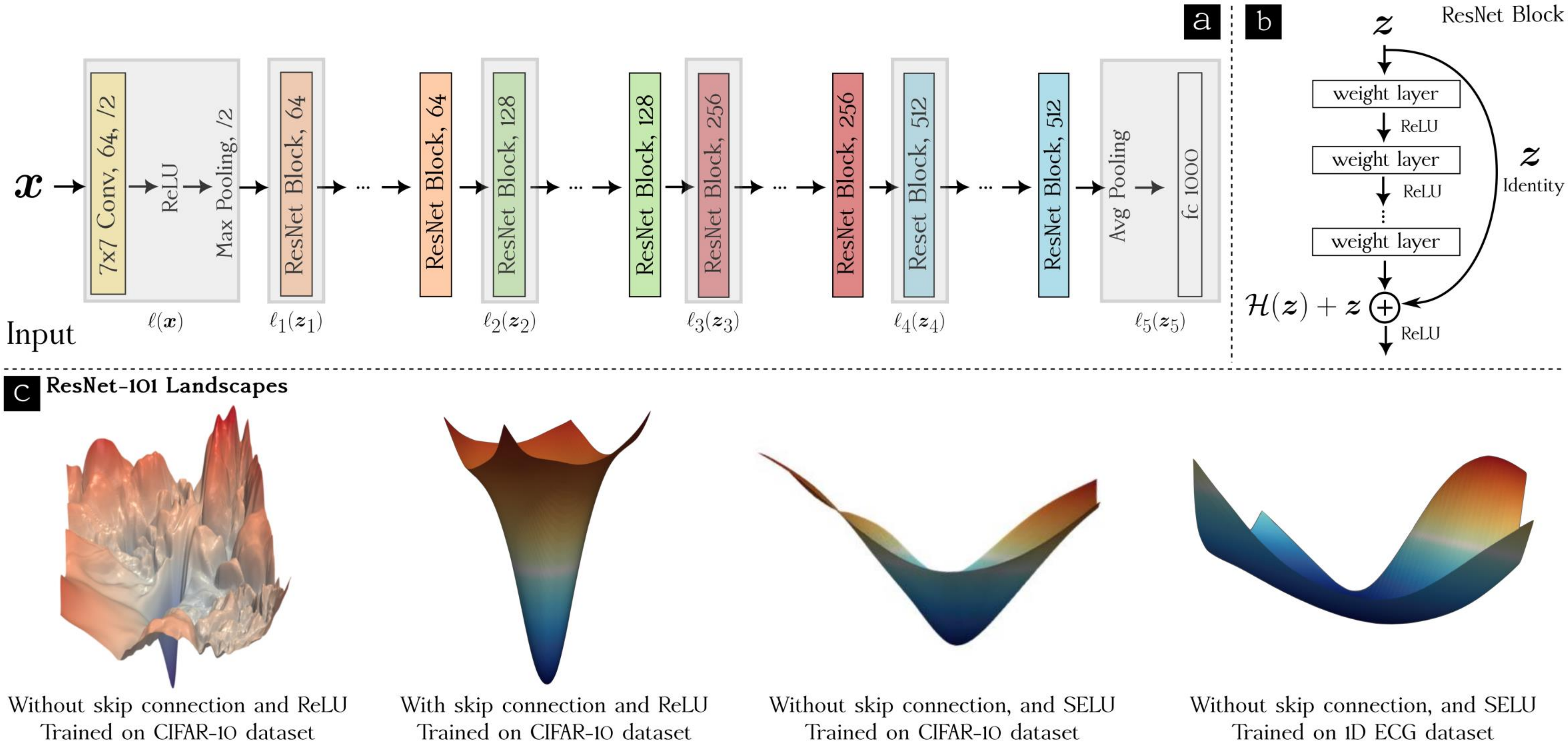


Generative Neural Networks



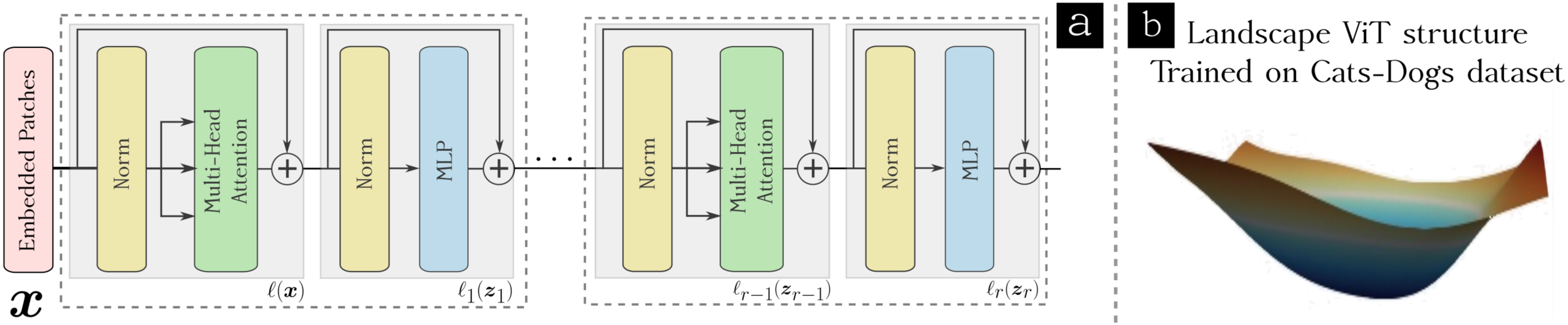
Pinilla, S., Sanabria, A., Bi, J., & Egiazarian, K. (2025). What Makes Neural Networks Trainable? Invexity as a Structural Design Principle in AI.

ResNet Structure



Pinilla, S., Sanabria, A., Bi, J., & Egiazarian, K. (2025). What Makes Neural Networks Trainable? Invexity as a Structural Design Principle in AI.

Transformer Structure



Theorem 8. *Let $f : \mathbb{R}^n \to \mathbb{R}^m$, and $g : \mathbb{R}^m \to \mathbb{R}^p$ be differentiable functions defined as*

$$f(\boldsymbol{x}) = \begin{bmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_m(\boldsymbol{x}) \end{bmatrix}, \quad g(\boldsymbol{z}) = \begin{bmatrix} g_1(\boldsymbol{z}) \\ \vdots \\ g_p(\boldsymbol{z}) \end{bmatrix},$$

for $f_i : \mathbb{R}^n \to \mathbb{R}$, and $g_j : \mathbb{R}^m \to \mathbb{R}$, with $p \leq m \leq n$, and $i = 1, \ldots, m$, $j = 1, \ldots, p$. If the Jacobian matrices $J_f(\boldsymbol{x}) \in \mathbb{R}^{m \times n}$, and $J_g(\boldsymbol{z}) \in \mathbb{R}^{p \times m}$ of functions f, and g respectively are full-row-rank for all $\boldsymbol{x} \in \mathbb{R}^n$, and $\boldsymbol{z} \in \mathcal{C}_f$, with $\mathcal{C}_f = \{\boldsymbol{z} \in \mathbb{R}^m | \exists \boldsymbol{x} \in \mathbb{R}^n$ such that $f(\boldsymbol{x}) = \boldsymbol{z}\}$, then functions

$$h_j(\boldsymbol{x}) = (g_j \circ f)(\boldsymbol{x}) = g_j(f(\boldsymbol{x})), \tag{12}$$

for $j = 1, \ldots, p$ are invex with respect to the same η.

Pinilla, S., Sanabria, A., Bi, J., & Egiazarian, K. (2025). What Makes Neural Networks Trainable? Invexity as a Structural Design Principle in AI.

References

- Bertsekas, D. P. (2012). Lecture slides on convex analysis and optimization based on 6.253 class lectures at the mass.
- Boyd, S., & Vandenberghe, L. (2004). Convex optimization. Cambridge university press.
- Klee, V. (1971). Convex Analysis (R. Tyrrell Rockafellar). SIAM Review, 13(2), 233-238.
- Bagirov, Adil, Napsu Karmitsa, and Marko M. Mäkelä. Introduction to Nonsmooth Optimization: theory, practice and software. Vol. 12. Cham, Heidelberg: Springer International Publishing, 2014.
- Pinilla, S., Mu, T., Bourne, N., & Thiyagalingam, J. (2022). Improved imaging by invex regularizers with global optima guarantees. Advances in Neural Information Processing Systems, 35, 10780-10794.

References

- Parikh, N., & Boyd, S. (2014). Proximal algorithms. Foundations and trends® in Optimization, 1(3), 127-239.
- Boyd, S., & Vandenberghe, L. (2004). Convex optimization. Cambridge university press.
- Beck, A. (2017). First-order methods in optimization. Society for Industrial and Applied Mathematics.
- Neal, P., Eric, C., Borja, P., & Jonathan, E. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® inMachine learning, 3(1), 1-122.
- Pinilla, S., & Thiyagalingam, J. (2024, May). Global optimality for non-linear constrained restoration problems via invexity. In The Twelfth International Conference on Learning Representations.

References

- Krull, A., Buchholz, T. O., & Jug, F. (2019). Noise2void-learning denoising from single noisy images. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2129-2137).
- Wen, F., Chu, L., Liu, P., & Qiu, R. C. (2018). A survey on nonconvex regularization-based sparse and low-rank recovery in signal processing, statistics, and machine learning. IEEE Access, 6, 69883-69906.
- Candès, E. J., & Wakin, M. B. (2008). An introduction to compressive sampling. IEEE signal processing magazine, 25(2), 21-30.
- Taken from https://www.spectroscopyeurope.com/article/hype-spectral-imaging
- Cai, Y., Lin, J., Lin, Z., Wang, H., Zhang, Y., Pfister, H., ... & Van Gool, L. (2022). Mst++: Multi-stage spectral-wise transformer for efficient spectral reconstruction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 745-755).

Thanks