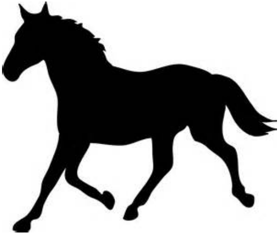
Image segmentation and classification using the Poisson Equation

Manuel Mende

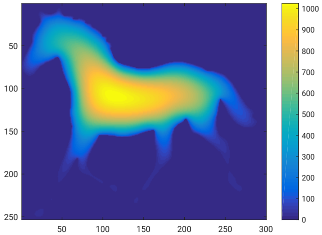
Shape Analysis

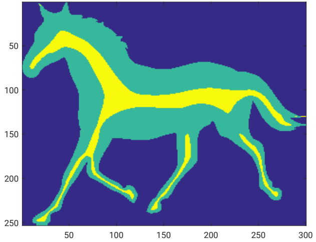
30.11.2016

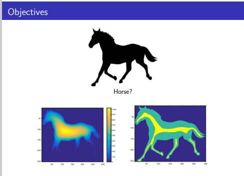
Objectives



Horse?



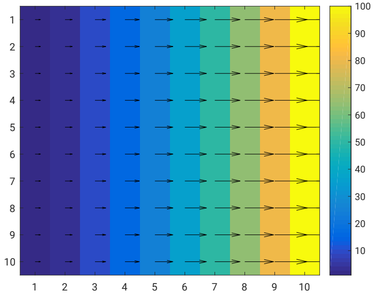




- Mathematical basics
  - Differentiation on $\mathbb{R}^2$
  - Random Walkers
- What is the Poisson Equation?
- Corner and Skeleton detection
- Classification using decision trees

Gradient

The *gradient* is defined for scalar functions and points into the direction of the greatest slope.

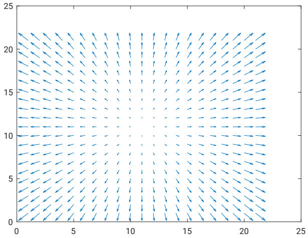


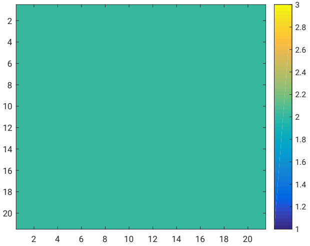
It can be calculated by partial differentiation:

$$grad\ f_{x,y} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} f_{x,y}$$

with f being a scalar field.

Divergence



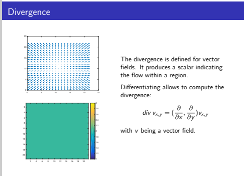


The divergence is defined for vector fields. It produces a scalar indicating the flow within a region.

Differentiating allows to compute the divergence:

$$div\ v_{x,y} = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})v_{x,y}$$

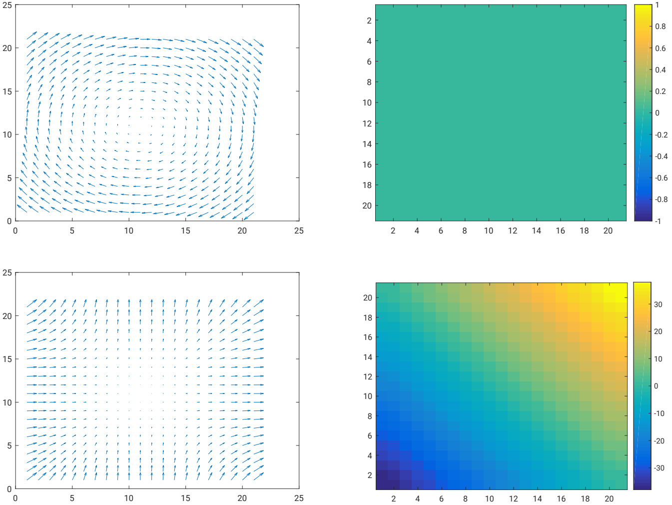
with v being a vector field.



Div describes the flow of values:

- We see a vector field
- div provides to any vector a scalar
- this scalar represents the flow in this region
- whether the point is source, sink or invariant
- We see: any point acts as a source (right/upper arrows are longer)
- differentiating computes the divergence

Divergence


0 5 10 15 20 25
2 4 6 8 10 12 14 16 18 20
1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1
30 20 10 0 -10 -20 -30

Laplace operator

The *Laplacian* Δ assigns the *divergence of the gradient* to any point of the function f:

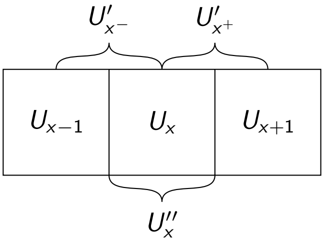
$$\Delta f = div(grad(f)) \tag{1}$$

while f can be differentiated twice and is a real-valued function. The result would be

$$\Delta f = \frac{\partial^2}{\partial x^2} f + \frac{\partial^2}{\partial y^2} f = f_{(xx)} + f_{(yy)} \tag{2}$$

Discrete differentiation (on $\mathbb{R}$)

Differentiation also works on a discrete mesh[1]



The first derivatives would be

$$U'_{x^-} = \frac{U_x - U_{x-h}}{h}$$

$$U'_{x^+} = \frac{U_{x+h} - U_x}{h}$$

The second differentiation step yields

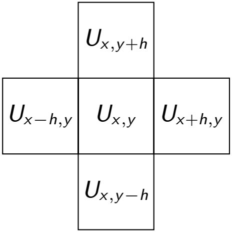
$$U''_x = \frac{U'_{x^+} - U'_{x^-}}{h} = \frac{U_{x+h} - 2U_x + U_{x-h}}{h^2}$$

[1] In this example only on $\mathbb{R}$

Discrete differentiation(on $\mathbb{R}^2$)

The same procedure can be applied for functions on $\mathbb{R}^2$:

$$\Delta U_{x,y} = \frac{U_{x+h,y} + U_{x-h,y} + U_{x,y+h} + U_{x,y-h} - 4U_{x,y}}{h^2}$$



which can be rewritten as

$$\Delta U_{x,y} = -\frac{4}{h^2} \cdot \left[U_{x,y} - \frac{1}{4} \left(U_{x+h,y} + U_{x-h,y} + U_{x,y+h} + U_{x,y-h} \right) \right] \tag{3}$$

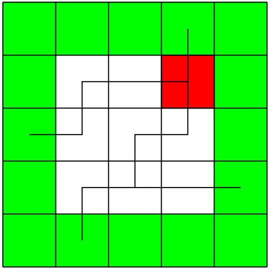
(3) is a discrete form of the Laplace operator (2): $\Delta f = f_{(xx)} + f_{(yy)}$

Random walker

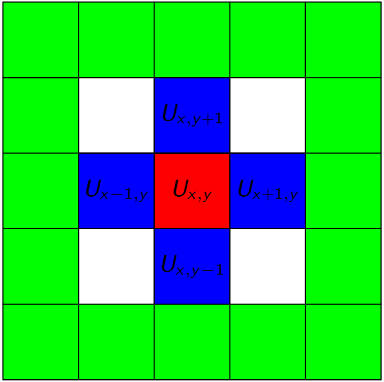
Random walkers execute random steps

- One random walk is executed recursively
- We start at a point and execute random steps until a boundary is hit
- Averaging various random executions leads to a measure
- Consider a shape S
- And a function $U(x, y)$ assigning this averaged value to any point in S
- The points at the boundaries – denoted by ∂S – satisfy $U(x, y) = 0$

This random walker leads to the *mean time to hit a boundary measure*



Mean time to boundary measure



- When we are on the boundary, the solution is zero
- Otherwise we can use probabilistic inference: We can visit each neighbour with a probability of $\frac{1}{4}$

$$U_{x,y} = h + \frac{1}{4}(U_{x+h,y} + U_{x-h,y} + U_{x,y+h} + U_{x,y-h}) \tag{4}$$

Mean time to boundary measure

Another representation of the *mean time to boundary measure* is

$$h = U_{x,y} - \frac{1}{4}(U_{x+h,y} + U_{x-h,y} + U_{x,y+h} + U_{x,y-h})$$

which can be rewritten by using the discrete laplacian, yielding

$$\Delta U_{x,y} = -\frac{4h}{h^2} \tag{5}$$

The Poisson Equation

The Poisson Equation is a differential equation which is defined as

$$-\Delta U = f$$

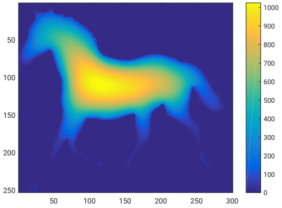
with the solution U and a function f. Especially (5)

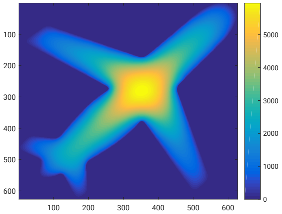
$$\Delta U_{x,y} = -\frac{4h}{h^2}$$

is an instance of this equation.

Properties of the Poisson Equation

- Optaining a solution requires boundary conditions, which are stated as $\forall(x, y) \in \partial S : U(x, y) = 0$
- The Level-Sets in this representation provide smoother versions of the boundaries
- Since many boundary points are considered – not only the euclidean distance – more global properties are available





Applications of the Poisson Equation

The solution to the Poisson Equation can be used to compute various helpful properties, among them

- Corners with concave regions on a shape
- Skeletons, the most central part of a shape

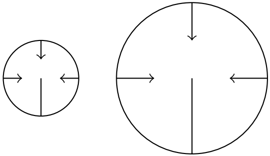
Corners

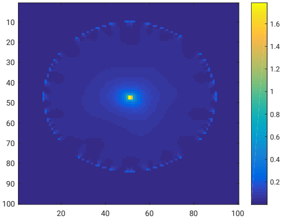
Corners occur at curved regions:

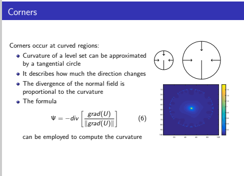
- Curvature of a level set can be approximated by a tangential circle
- It describes how much the direction changes
- The divergence of the normal field is proportional to the curvature
- The formula

$$\Psi = -div \left[\frac{grad(U)}{\|grad(U)\|} \right] \tag{6}$$

can be employed to compute the curvature

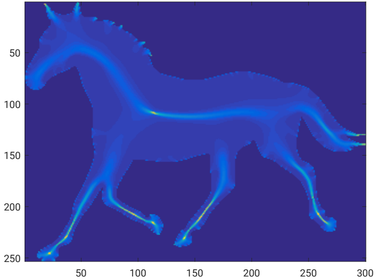




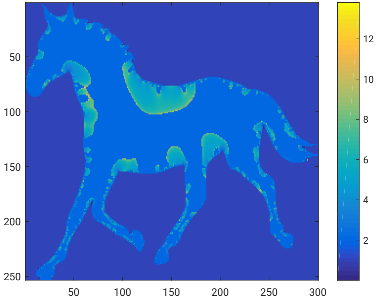


The curvature describes, how much the direction of a curve changes. This can be accomplished by calculating the divergence of the normal field on the curve. Since the normal field of the curve is rectangular to the curve and $(\partial x, \partial y) \times v_{set}$ is the trace of the hessian – which sum is independent of the koordinate system – we can simply compute the laplacian at any point in the original koordinate system and optain the cuvature

Real Corners



$\Psi_{x,y} > 0$

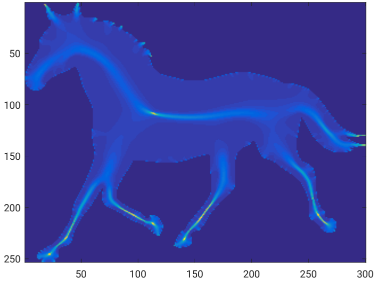


$log(-\Psi)$

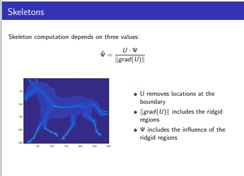
Skeletons

Skeleton computation depends on three values:

$$\tilde{\Psi} = \frac{U \cdot \Psi}{\|grad(U)\|}$$



- U removes locations at the boundary
- $\|grad(U)\|$ includes the ridgid regions
- Ψ includes the influence of the ridgid regions

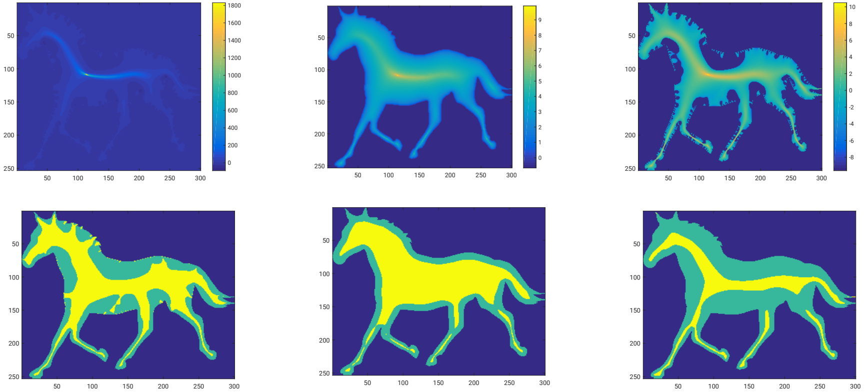


Influence of the terms:

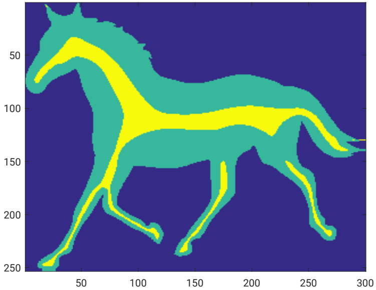
- Since the values of U are small there
- Since the gradient is very low
- Since the value for ridgid regions is highly positive, especially because of the division by the abs grad value

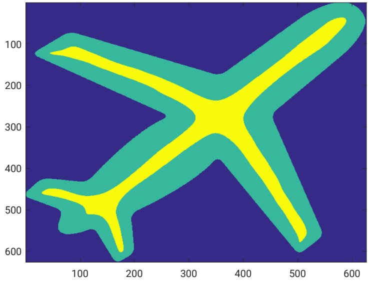
Influence of the terms

$$\tilde{\Psi} = \frac{U \cdot \Psi}{\|grad(U)\|}$$



Real Skeletons





Skeletons computed with $\tilde{\Psi}$ and thresholded with the mean values of the shape.

Classification

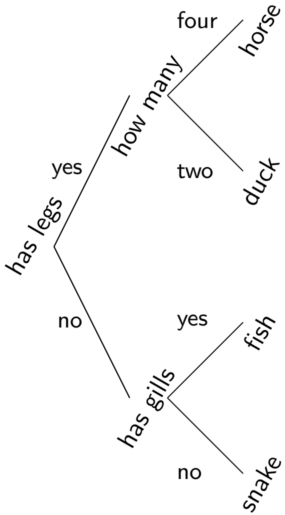
This properties can be used to compute features/measures that can be used to classify shapes

Decision trees

Binary decision trees

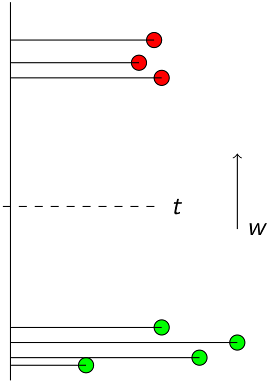
- represent knowledge about a domain
- can be used to infer properties about unknown objects, e.g. a label for a shape
- are derived by using a trainings set

Each edge represents a set of features, each leaf represents a class.

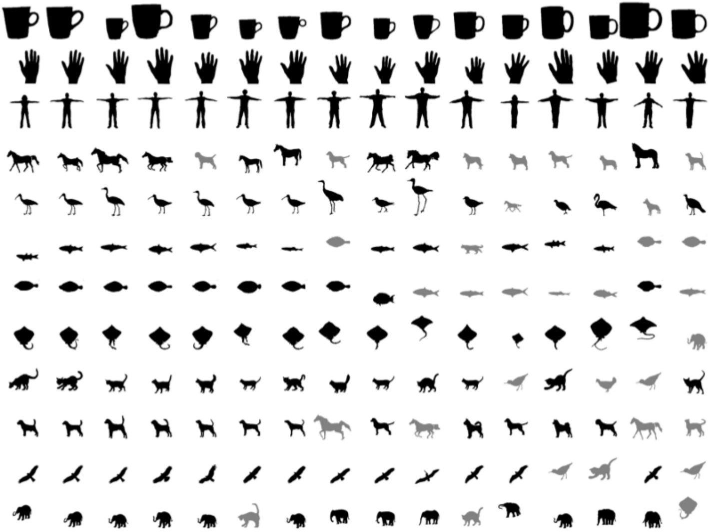


Training the classifier

- Each sample is represented by a high dimensional feature vector
- Splitting is achieved by projecting data onto a line
  and
- Applying a threshold to separate the data
- The ideal combination of both – direction w and threshold t – splits the data
- A split is considered good, if the mixture in the subsets is reduced

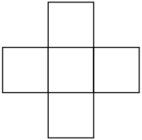


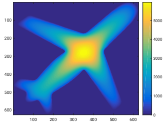
Results of Classification

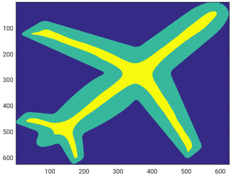


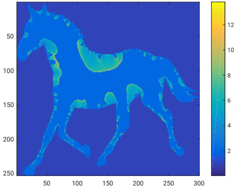
Conclusion

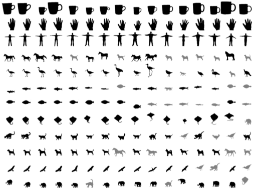
- There is a discrete form of the Laplace operator
- It can be used to solve the Poisson Equation on a discrete grid
- This solution is equivalent to a random walker measure
- It can be used to compute interesting properties of a shape
- Those properties can be used to classify shapes using decision trees











Thank you for your attention!