REVISITING THE K-MEANS ALGORITHM FOR FAST TRAJECTORY SEGMENTATION

Luis A. Leiva, Enrique Vidal <luileito,evidal AT iti.upv.es>

SIGGRAPH2011 VANCOUVER

WHAT?

A novel trajectory segmentation technique based on the K-means algorithm.

WHY?

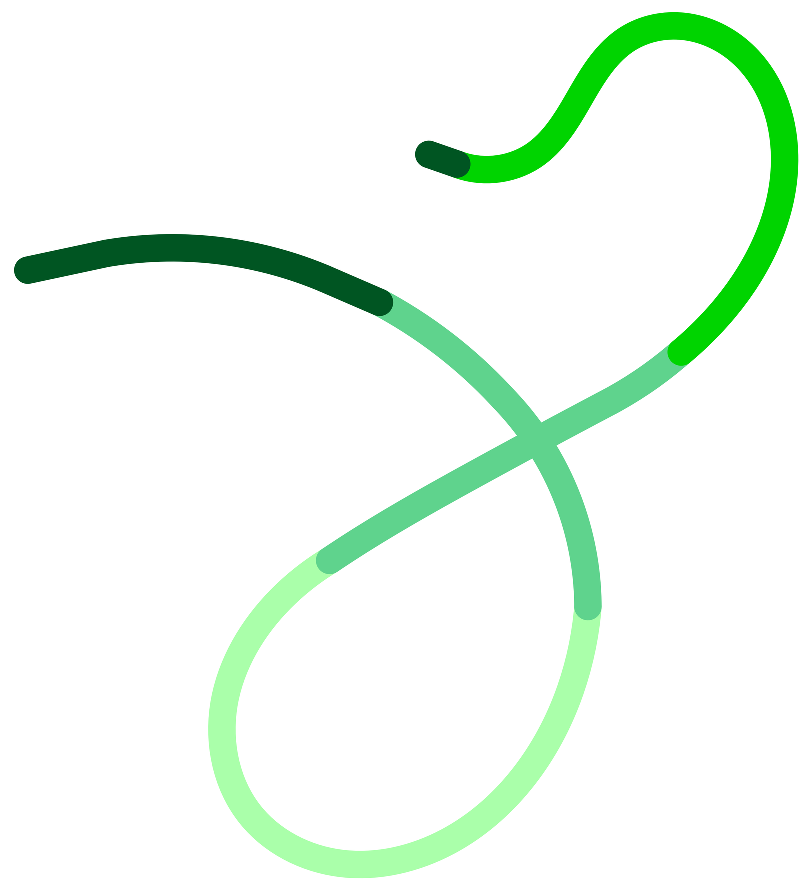
Leverage K-means strengths (simplicity, fast convergence) minimizing drawbacks (initialization, instance order).

HOW?

Initialize data with Trace Segmentation [2] and impose temporal constraints to the sequential version of K-means [1].

AN ILLUSTRATED 2D EXAMPLE

(a) Original Trajectory
480 points @ 30 fps



(b) Classical K-means
50 ms, 30 iterations

(c) Our algorithm
10 ms, 3 iterations

FEATURES

Accurate: it guarantees the convergence to a minimum-distortion segmentation;

Robust: each run for a given K always yields in the same result;

Fast: only two clusters are inspected in each classification step;

Near-linear complexity: $\Theta(kd)$ instead of $\Theta(nkd)$;

Not extra input parameters: just the same as in K-means;

Real-time, large DBs: the number of data points does influence when updating the centroids;

On-line learning: clusters can be updated while new samples arrive without affecting the previous data structure.

TRACE SEGMENTATION

Input: Trajectory $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$; Traces $M \geq 2$
Output: Normalized trajectory $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_M$

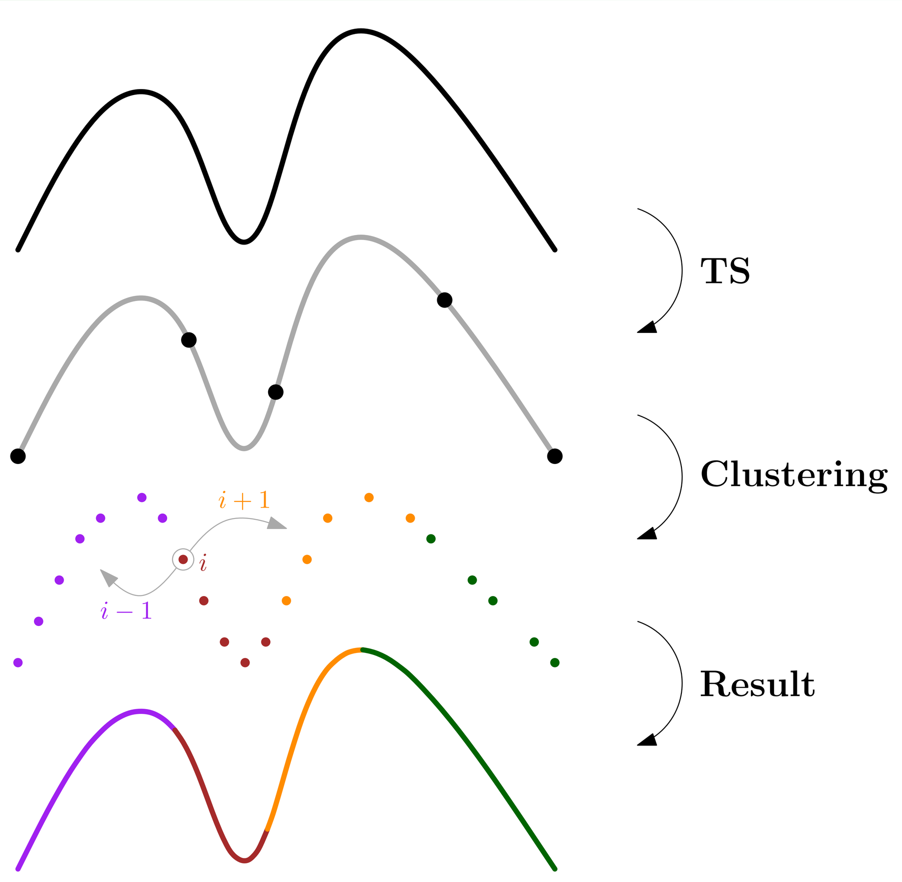
$L_1 = 0$ `// accumulated trace length`
for $n = 2$ to N **do**
  $L_n = L_{n-1} + \| \boldsymbol{x}_n - \boldsymbol{x}_{n-1} \|$
$\lambda = \frac{L_N}{M-1}$ `// segment length`
$\boldsymbol{y}_1 = \boldsymbol{x}_1$ `// first point`
$n = 2$
for $m = 2$ to $M-1$ **do** `// interpolate`
  while not $L_{n-1} \leq (m-1)\lambda \leq L_n$ **do**
    $n{+}{+}$
  $\boldsymbol{y}_m = \boldsymbol{x}_{n-1} + (\boldsymbol{x}_n - \boldsymbol{x}_{n-1}) \frac{(m-1)\lambda - L_{n-1}}{L_n - L_{n-1}}$
$\boldsymbol{y}_M = \boldsymbol{x}_N$ `// last point`

SEQUENTIAL CLUSTERING

Input: Trajectory X; No. clusters C
Output: Partition $\prod^*$; Centroids $\boldsymbol{\mu}_c$; Distortion J

$\prod^0 = \{X_1, \ldots, X_C\}$ `// use trace segmentation`
for $c = 1$ to C **do**
  $\boldsymbol{\mu}_c = \frac{1}{n_c} \sum_{\boldsymbol{x} \in X_c} \boldsymbol{x}$ `// means`
  $J = J + \sum_{\boldsymbol{x} \in X_c} \| \boldsymbol{x} - \boldsymbol{\mu}_c \|^2$ `// SSEs`
while $transfers$ **do**
  $transfers = false$
  for all $i : \boldsymbol{x} \in X_i$ **do**
    if $n_i > 1$ **then** `// classify`
      $j^* = \underset{i-1<j<i+1}{\arg\min} \frac{n_j}{n_j+1} \| \boldsymbol{x} - \boldsymbol{\mu}_j \|^2$
      $\Delta J = \frac{n_{j^*}}{n_{j^*}+1} \| \boldsymbol{x} - \boldsymbol{\mu}_{j^*} \|^2 - \frac{n_i}{n_i-1} \| \boldsymbol{x} - \boldsymbol{\mu}_i \|^2$
      if $\Delta J < 0$ **then** `// reallocate`
        $transfers = true$
        $\boldsymbol{\mu}_i = \boldsymbol{\mu}_i - \frac{\boldsymbol{x} - \boldsymbol{\mu}_i}{n_i - 1}$ $\boldsymbol{\mu}_{j^*} = \boldsymbol{\mu}_{j^*} + \frac{\boldsymbol{x} - \boldsymbol{\mu}_{j^*}}{n_{j^*}+1}$
        $X_i = X_i - \{\boldsymbol{x}\}$ $X_{j^*} = X_{j^*} + \{\boldsymbol{x}\}$
        $J = J + \Delta J$

STEPS EXPLAINED



INTERACTIVE PROTOTYPE

http://personales.upv.es/luileito/wkm/siggraph-demo.html

REFERENCES

[1] DUDA, R. O., HART, P. E., AND STORK, D. G. 2001. *Pattern Classification*, 2nd ed. John Wiley & Sons, ch. Unsupervised Learning and Clustering, 517–599.

[2] KUHN, M. H., TOMASCHEWSKI, H., AND NEY, H. 1981. Fast nonlinear time alignment for isolated word recognition. In *Proc. ICASSP*, 736–740.

ACKNOWLEDGEMENTS

Work supported by the Spanish research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018).