Progress in Dynamic Texture Showcase

Sándor Fazekas  Dmitry Chetverikov

Computer and Automation Research Institute
Geometric Modelling and Computer Vision Lab
Budapest, Hungary
visual.ipan.sztaki.hu

Showcase Meeting, Budapest, 23-April-2007
1. Visual motion and dynamic texture

2. Dynamic texture detection
   - Regular and nonregular optical flows
   - Segmentation using regular and nonregular flows

3. Results

4. Available versions of algorithm

5. Summary
Categories of visual motion patterns

- **Activities**
  - periodic in time, localised in space
    ⇒ walking, digging

- **Motion events**
  - no temporal or spatial periodicity
    ⇒ opening a door, jump

- **Temporal textures**
  - statistical regularity, indeterminate spatial and temporal extent
    ⇒ fire, smoke
Examples of dynamic textures

⇒ show sample videos
DynTex database by NoE MUSCLE

- 656 digital videos
- PAL 720 x 576, 25 fps
- Length ≥ 250 frames
- Closeups and contexts
- Static/moving camera
- Indoor and outdoor natural scenes
- Annotated, categorised (work in progress)
- Available on the Web (> 50 registered users)

http://www.cwi.nl/projects/dyntex/

(In collaboration with R. Péteri, M. Huiskes, and CWI)
Work in progress with Tel-Aviv University (TAU)
- Tomer Amiaz
- Nahum Kiryati

Dynamic textures have strong **intrinsic dynamics**
- motion cannot be compensated by shift/rotation
- **intensity constancy** assumption not valid
- standard (regular) optical flow not precise

Use **intensity conservation** assumption instead
- non-regular optical flow with divergence term
- intensity may diffuse

Dynamic texture detection
- segmenting flow into regular and non-regular part
- indicator function in level-set implementation
Brightness conservation assumption

- **Non-regular optical flow** (compared to Horn-Schunck)

  Brightness constancy: \( l(x + u, y + v, t + 1) = l(x, y, t) \)
  
  Optical flow constraint: \( l_t + ul_x + vl_y = 0 \)

  Brightness conservation: \( l(x + u, y + v, t + 1) = l(x, y, t)(1 - u_x - v_y) \)
  
  Continuity equation: \( l_t + ul_x + vl_y = -l \cdot (u_x + v_y) \)

- Brightness of an image point (in one frame) can propagate to its neighborhood (in the next frame)
- Captures more information than a regular flow
- Encodes the warp residual of a regular flow
- Applicable to strong dynamic textures (generic feature)
Optical flow equations

- Horn-Schunck
  - brightness constancy ($\mathbf{v} = (u, v)$: velocity vector)
    \[
    \partial_t l + \mathbf{v} \cdot \nabla l = 0
    \]

- Lagrangian
  \[
  L_{HS}(u, v) = (l_t + ul_x + vl_y)^2 + \alpha(u_x^2 + u_y^2 + v_x^2 + v_y^2)
  \]
  - minimise $F_{HS}(u, v) = \int L_{HS}(u, v) \, dxdy$

- Brightness conservation
  \[
  \partial_t l + \mathbf{v} \cdot \nabla l + l \ \text{div} \mathbf{v} = 0
  \]
  - Lagrangian more complicated, but essentially similar
More precise motion compensation by nonregular flow

(a,d): frame 1 of dynamic texture; (b,e): frame 2 warped back by regular flow; (c,f): same by non-regular flow
Segmentation as a variational problem

\[
L_{\text{DTS}}(u, v, \tilde{u}, \tilde{v}, \phi) = \\
(l_t + ul_x + vl_y)^2 H(\phi) + (l_t + \tilde{u}l_x + \tilde{v}l_y + l\tilde{u}_x + l\tilde{v}_y)^2 H(-\phi) \\
+ \alpha(u_x^2 + u_y^2 + v_x^2 + v_y^2) + \tilde{\alpha}(\tilde{u}_x^2 + \tilde{u}_y^2 + \tilde{v}_x^2 + \tilde{v}_y^2) + \beta(\tilde{u}^2 + \tilde{v}^2) \\
+ \nu|\nabla H(\phi)|
\]

\[
F_{\text{DTS}}(u, v, \tilde{u}, \tilde{v}, \phi) = \int L_{\text{DTS}}(u, v, \tilde{u}, \tilde{v}, \phi) \, dx \, dy
\]

- Brightness constancy on static and weak dynamic regions
- Brightness conservation on strong dynamic regions
- Smooth boundary of segmented regions
- Solved (Euler-Lagrange eqs., discretisation based on central derivatives, iterative solver, . . . )
Results

⇒ show sample videos
Versions: Making it faster

- Full algorithm
  - precise segmentation
  - no thresholding needed (decision by indicator function)
  - currently, slow (15–20 sec/frame)
  - ⇒ make faster using graph cuts

- Fast simplified version
  - less precise segmentation
  - threshold learned, then adjusted adaptively
  - close to real-time (5–10 fps)

- Real-time simplified version
  - less precise segmentation, sometimes errs
  - threshold adjusted adaptively
  - real-time (20–25 fps)
Collaborative work of SZTAKI and TAU
Showcase with Bilkent
Generic method for detecting dynamic textures
  processes of various physical origin
More than just detection/segmentation
  calculates optical flow useful for recognition
Plans
  speed up full algorithm (graph cuts)
  improve real-time version: automatic threshold, adaptivity
  distinguish between DTs and other fast motion
  integrate with periodicity detection and DT recognition