INTERACTIVE OBJECT SEGMENTATION WITH NOISY BINARY INPUTS

Gregory Canal*, Sivabalan Manivasagam*, Shaoheng Liang†, Christopher Rozell*

*Georgia Institute of Technology, †Tsinghua University

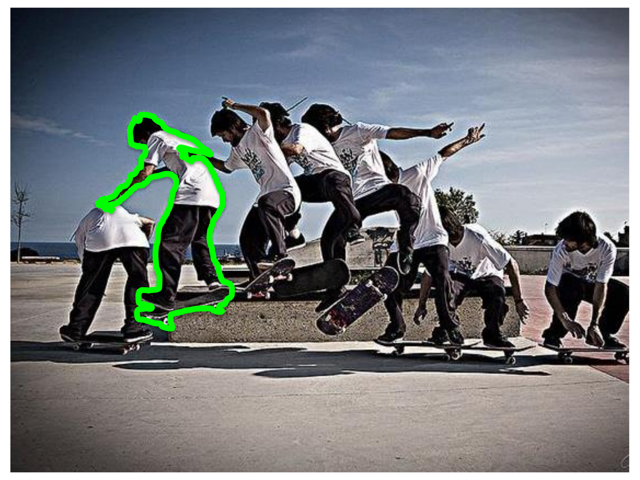
Summary

In many image segmentation settings where users interact with a computer using only limited, low-complexity inputs, segmentation algorithms must remain robust to input errors while outputting segments specified by the user as efficiently and accurately as possible. We study the scenario of segmentation of real-world *objects* in images specified by a user with *noisy, binary* inputs. We achieve this by modeling an ellipse circumscribing the user's desired segment as a *message* in a *communications channel with feedback* and extending an optimal algorithm in feedback information theory. We compare our algorithm (EllipseLex) against a baseline method in simulation and demonstrate improved performance for segmenting objects in images.

Main result: simple, optimal algorithm for interactive object segmentation with noisy, binary inputs

Interactive object segmentation

Specify real-world object region of interest in image using only **noisy, low-complexity inputs**



Example scenarios

- hands-free operating room (foot pedal switch) (Dubost et al. 2016)
- repetitive strain injury prevention (Sadeghi et al. 2009)
- brain-computer interfacing for paralyzed users (Wolpaw et al. 2002)
- non-traditional motor commands (tongue drive) (Park et al. 2012)

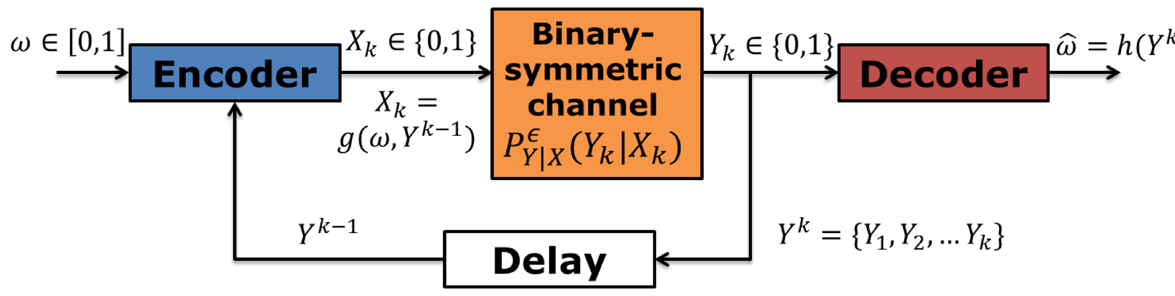
Ellipse selection with binary inputs

- specify ellipse masking desired object
- using only binary inputs
- subject to binary noise (i.i.d. symmetric bit flips)

Original EllipseLex Post-processed

- ➢ ellipse masks can circumscribe many real-world objects
- ➢ mask can be enhanced with any bounding box post-processing algorithm (e.g. GrabCut) (Rother et al. 2004)

Noisy channel with feedback



Interactive model: feedback communications system

- **Message** ω: desired ellipse
- **Encoder**: user issuing binary inputs
- **Binary-symmetric channel** (BSC): noisy channel with chance of input bit flip i.e. crossover probability of ϵ
- **Decoder**: desired ellipse guessed as $\widehat{\omega}$
- **Delay**: channel outputs observable by encoder

Optimal communication with feedback

Posterior matching encoding / decoding scheme:

- simple, intuitive encoding rule
- only needs to compare guess ($\widehat{\omega}$) and intent (ω)
- efficient in the number of inputs
- robust to bit flip errors

$$\widehat{\omega} = \operatorname{median}(F_{\omega|Y^{k-1}}) \qquad X_k = \begin{cases} 1, & \omega \geq \widehat{\omega} \\ 0, & \text{else} \end{cases}$$

$$P\left(|\omega - \widehat{\omega}| > 2^{-kR}\right) \to 0 \text{ as } k \to \infty, \text{ i.e. } \widehat{\omega} \to \omega$$

for any $R < C$ (channel capacity)

Discrete message set: Burnashev-Zigangirov (BZ) algorithm

- ➢ posterior matching with $\widehat{\omega}$ set to approximate median of posterior distribution over ordered message set

(Shayevitz & Feder 2011, Coleman 2009, Omar et al. 2010, Akce et al. 2010, Castro & Nowak 2009)

Application via lexicographical ordering

Construct a lexicon $\boldsymbol{Z}$ of ellipse masks

- lexicographical ordering: two ellipses $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ can be sorted (or alphabetized)
- each ellipse has letters from distinct alphabets
- alphabetize by comparing first letter that differs between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$
- use ellipse lexicon as ordered message set in BZ

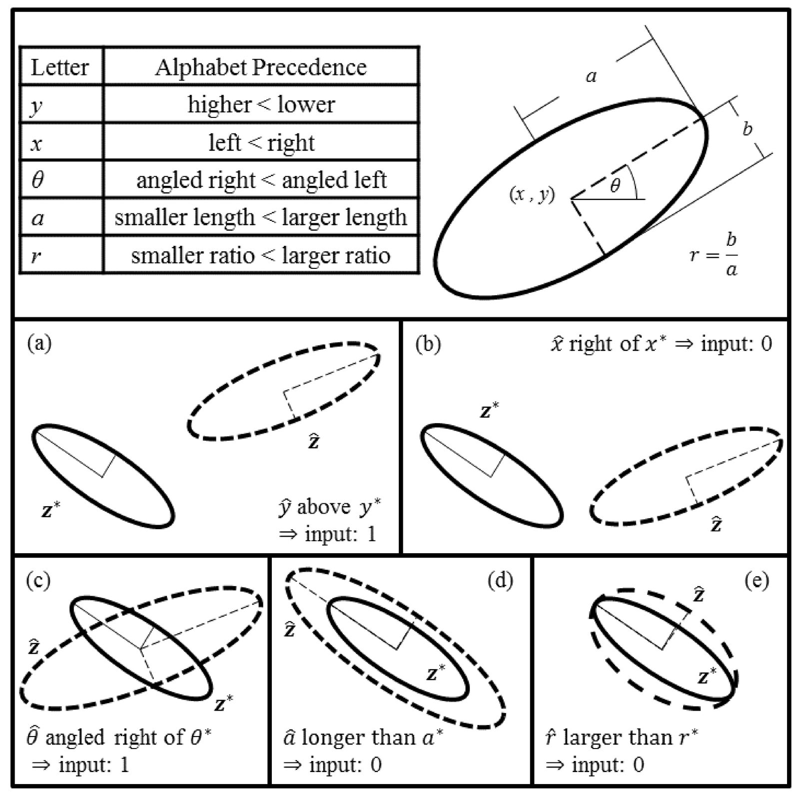
Ellipse letters:

1) vertical position: y
2) horizontal position: x
3) angle: θ
4) major axis length: a
5) minor-to-major axis aspect ratio: r

Algorithm 1: EllipseLex

Input: target ellipse mask $\mathbf{z}^* = (y^*, x^*, \theta^*, a^*, r^*)$
  $\hat{\mathbf{z}}_0 \leftarrow$ initial ellipse guess
  for $k \leftarrow 1$ **to** K **do**
    if $\hat{y}_{k-1} \neq y^*$ **then**
      compare vertical position y
    else if $\hat{x}_{k-1} \neq x^*$ **then**
      compare horizontal position x
    else if $\hat{\theta}_{k-1} \neq \theta^*$ **then**
      compare angle θ
    else if $\hat{a}_{k-1} \neq a^*$ **then**
      compare major axis length a
    else
      compare aspect ratio r
    end if
    let ρ represent the letter to be compared
    if $\rho^* \geq \hat{\rho}_{k-1}$ **then**
      $X_k = 1$ input 1
    else if $\rho^* < \hat{\rho}_{k-1}$ **then**
      $X_k = 0$ input 0
    end if
    $Y_k = BSC(X_k, p)$ BSC with crossover p
    $\hat{\mathbf{z}}_k = BZ(Y_k)$ update posterior, return median
  end for
Output: $\hat{\mathbf{z}}_K$

Construction of ellipse lexicon



- ➢ user implicitly envisions ellipse mask around desired object
- ➢ steers current guessed ellipse towards object with BZ algorithm and lexicon rules
- ➢ ellipse progresses from coarse to fine approximation of object

Comparison against prior work

- ➢ Comparison against 'N-Questions' baseline (Rupprecht et al. 2015)
  - ➢ active, pixelwise '20 Questions' style game: *is my guessed pixel in your region of interest?*
  - ➢ performs well on *arbitrary* regions of interest

Row 1: source image

Row 2: ground-truth segment

Row 3: EllipseLex (no post-processing)

Row 4: N-Questions

Experimental results

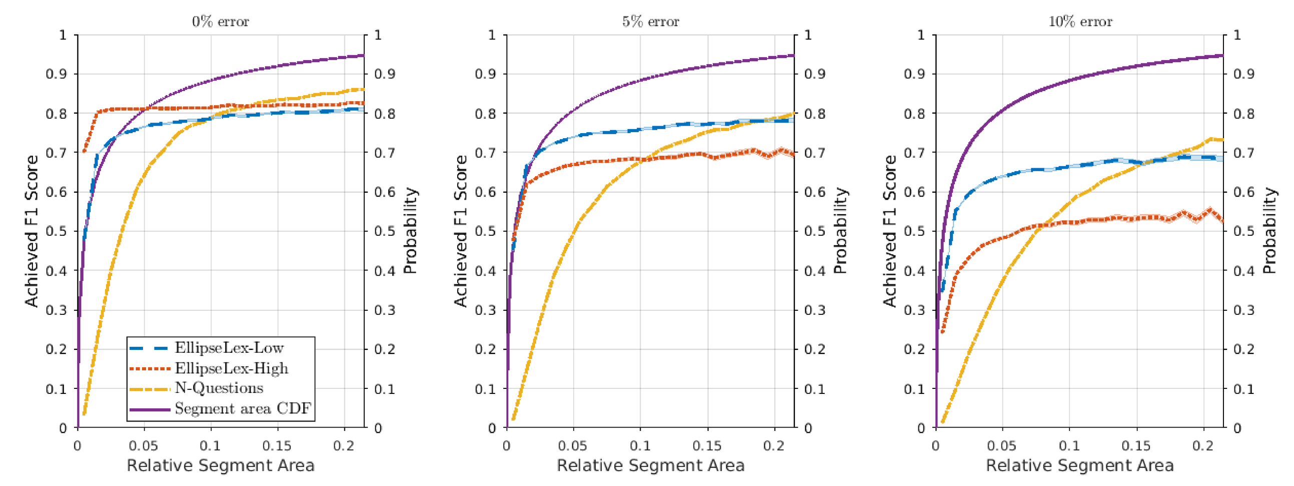
- Simulated runs using MS-COCO object segments (Lin et al. 2014)
- Segmentation performance metric: *F1 score* (Dice 1945)
  - harmonic mean of *precision* and *recall*
  - between 0 and 1, higher scores better
  - F1 score compared after K inputs
- i.i.d. binary noise at fixed crossover probability
  - 0% (noiseless), 5%, 10% error
- Low/High lexicon resolution
  - EllipseLex-Low alphabet sizes: (15,20,10,20,10)
  - EllipseLex-High alphabet sizes: (100,100,20,20,10)

Analysis 1: F1 versus number of inputs

Method	Noise	$K = 10$	$K = 20$	$K = 30$
EllipseLex-Low	0%	**0.2050**	**0.5927**	0.5949
EllipseLex-High	0%	0.1886	0.4400	**0.7487**
N-Questions	0%	0.1462	0.2112	0.2569
EllipseLex-Low	5%	**0.1720**	**0.4472**	**0.5685**
EllipseLex-High	5%	0.1353	0.2482	0.5518
N-Questions	5%	0.1269	0.1747	0.2064
EllipseLex-Low	10%	**0.1379**	**0.3021**	**0.4616**
EllipseLex-High	10%	0.0970	0.1766	0.3344
N-Questions	10%	0.1145	0.1476	0.1686

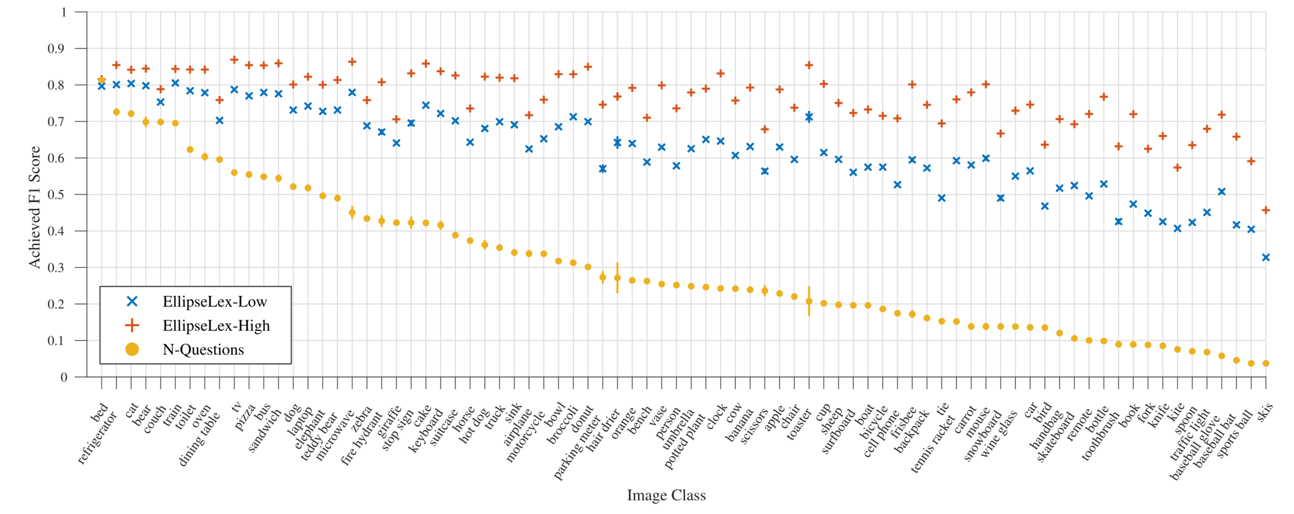
- ➢ EllipseLex outperforms N-Questions
  - ➢ EllipseLex-Low: faster F1 growth, noise resilient
  - ➢ EllipseLex-High: greater achieved F1 in noiseless setting

Analysis 2: F1 versus segment area (30 inputs)



- ➢ higher EllipseLex performance on small/medium segments
- ➢ higher N-Questions performance on large segments

Analysis 3: F1 versus object category (30 inputs)



- ➢ EllipseLex outperforms N-Questions
- ➢ EllipseLex-High achieves highest specificity

Main finding: EllipseLex is a simple, efficient, and robust method for interactively specifying real-world objects in images with noisy, binary inputs