Quantifying Copyright

James Grimmelmann

August 11, 2017

I. Information Theory

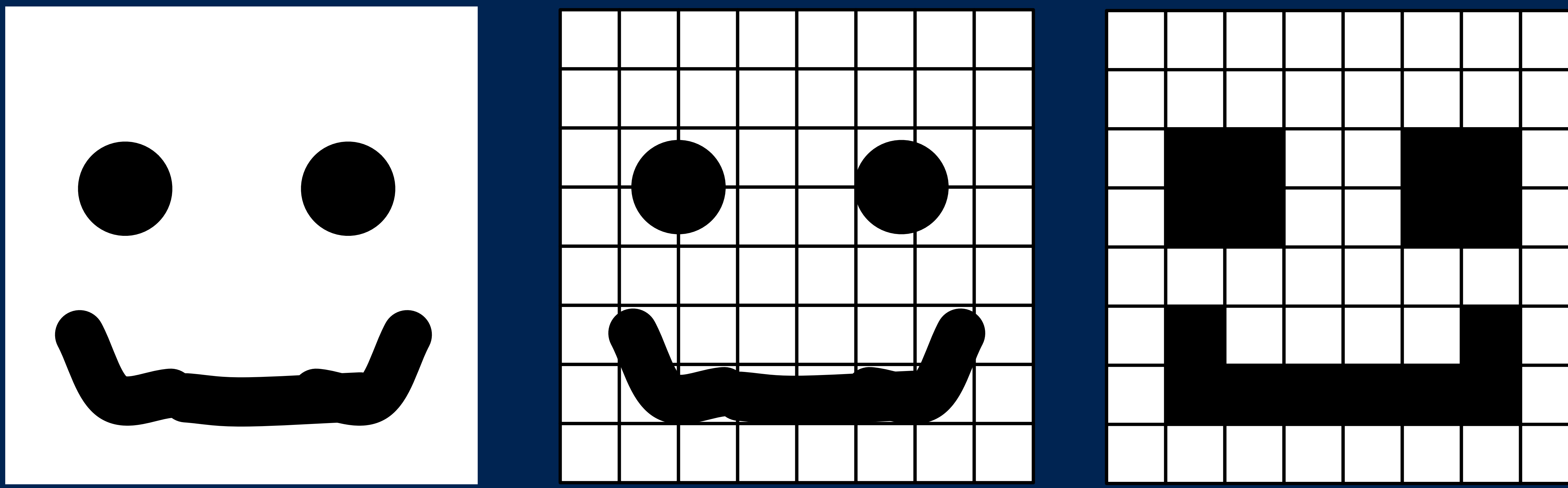
Three key ideas

- Digital encoding
- Counting bits
- Compression

1. Digital encoding

H	e	l	l	o	!
72	101	108	108	111	33
01001000	01100101	01101100	01101100	01101111	00100001

1. Digital encoding

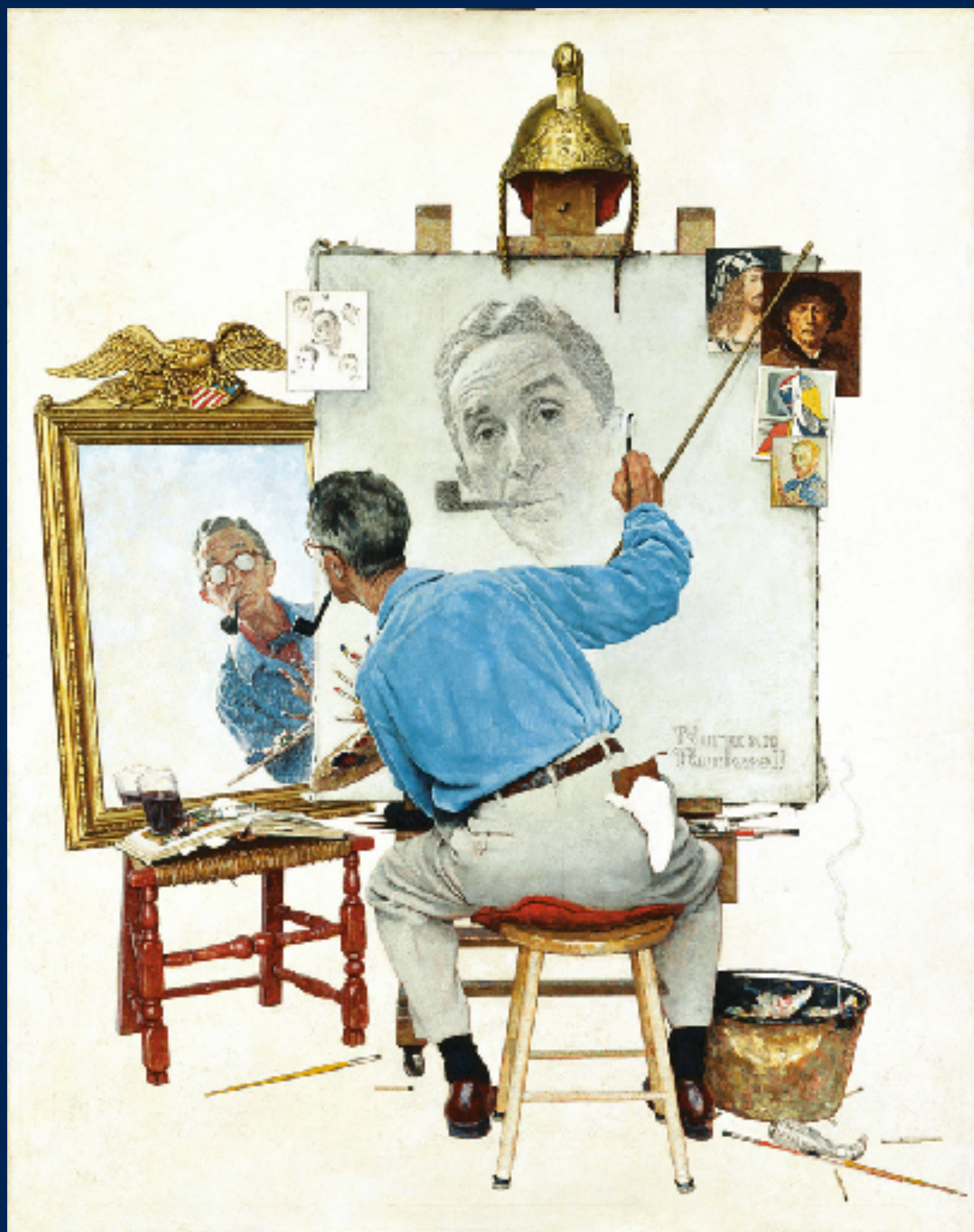


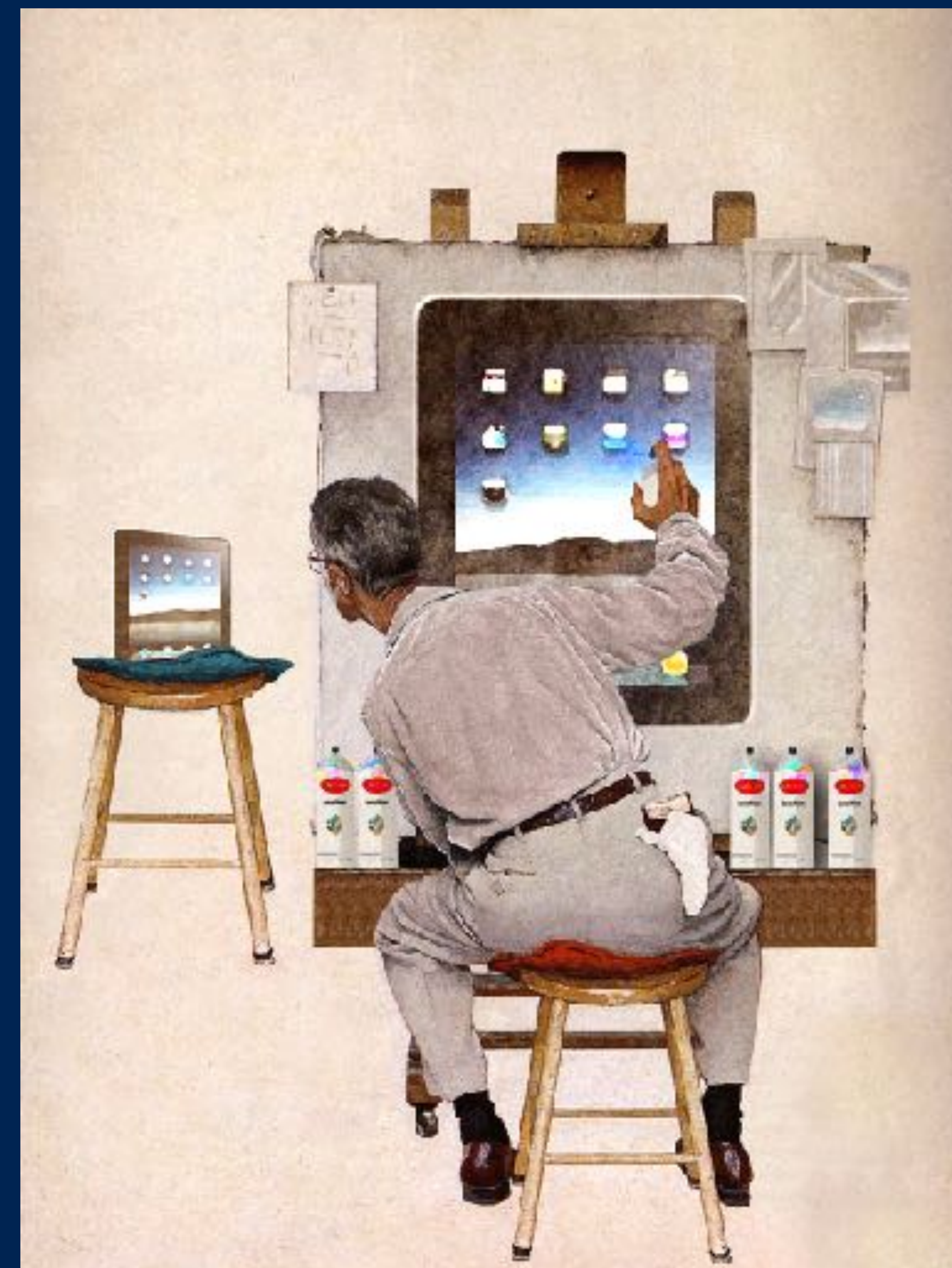
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	0	0	1	1	0	0	1
1	0	0	1	1	0	0	1
1	1	1	1	1	1	1	1
1	0	1	1	1	1	0	1
1	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1

1. Digital encoding




1. Digital encoding





2. Counting bits

`You Got the Right One, Uh-Huh`

232 bits

2. Counting bits

```
Thou still unravished bride of quietness,
Thou foster-child of silence and slow time,
Sylvan historian, who canst thus express
A flowery tale more sweetly than our rhyme
…
“Beauty is truth, truth beauty,-that is all
Ye know on earth, and all ye need to know”
```

17,544 bits

3. Compression

English	UTF-8	WTF-8
A	01000001	010000010000000000000000 000000000000000000000000
B	01000010	010000100000000000000000 000000000000000000000000
C	01000011	010000110000000000000000 000000000000000000000000
D	01000100	010001000000000000000000 000000000000000000000000

3. Compression

```
Happy_birthday_to_you←
Happy_birthday_to_you←
Happy_birthday_dear_X←
Happy_birthday_to_you←
```

704 bits

3. Compression

```
1:Happy_birthday
2:*1_to_you←

*2*2*1_dear_X*2
```

368 bits

II. Kolmogorov Complexity

Two branches of information theory

- (Shannon) information theory: communications, processes, encodings, noise
  - Properties of *communication systems*
  - *Cf.* Fromer, Scafidi
- (Algorithmic) information theory: computation, compressibility, encodings
  - Properties of *individual texts*

The Kolmogorov complexity (K) of a work (x) is the shortest (min) length (l) of a program (p) which produces ($\rightarrow$) the work

$$K(x) = \min_{p \rightarrow x} l(p)$$

A work is only as complex as its shortest encoding

Naive encodings

XXXXXXXXXX (10 Xs)	80 bits
XXXXXXXX…XXXXXX (100 Xs)	800 bits
XXXXXXXX…XXXXXXXX (1000 Xs)	8000 bits
XXXXXXXX…XXXXXXXXXX (10000 Xs)	80000 bits

Sophisticated encodings

```
XXXXXXXXXX (10 Xs)                         80 bits
for($i=0;$i<10;$i++){print"X"}            240 bits

XXXXXXXX…XXXXXX (100 Xs)                  800 bits
for($i=0;$i<100;$i++){print"X"}           248 bits

XXXXXXXX…XXXXXXXX (1000 Xs)              8000 bits
for($i=0;$i<1000;$i++){print"X"}          256 bits

XXXXXXXX…XXXXXXXXXX (10000 Xs)          80000 bits
for($i=0;$i<1000;$i++){print"X"}          264 bits
```

Comparing works

Kolmogorov complexity

$$K(x) = \min_{p \to x} l(p)$$

Conditional Kolmogorov complexity

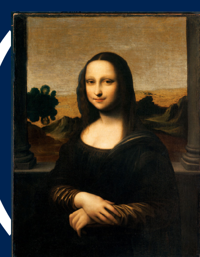
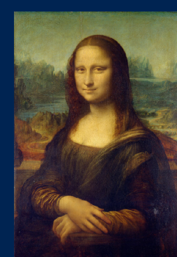
$$K(x|y) = \min_{p(y) \to x} l(p)$$

$K(x)$ = complexity of x

$K(x|y)$ = complexity of x not due to y

$K(x) - K(x|y)$ = complexity of x due to y

K() = high

K( | ) = medium

K() - *K*( | ) = high

K( | ) = high

K() - *K*( | ) = low

III. Copyright

Tentative idea #1: counting arguments

- *Feist:* there are (250,000,000!)/(50,000!)(249,950,000!) ways to select 50,000 listings
  - ~1,500,000 bits to describe an arbitrary selection
- There are 50,000! ways to arrange those listings
  - ~750,000 bits to describe an arbitrary ordering
- But the *actual* selection and arrangement and require far fewer bits: too few for a copyright

Tentative idea #2: quantify factor three

- *Perfect 10*: how much of a 1000x1000 image does a 100x100 thumbnail copy?
    - 100%, because it's the "whole" image?
    - 1%, because it has 1/100 as many pixels?
- Why not compare .JPG file sizes?

Other tentative ideas

- Rule-based creativity: can't extract more bits of expression than you put in
- Merger kicks in when there are "only a limited number of ways" to an express an idea
- Scènes à faire are about predictability: in a hard-boiled detective novel, it adds almost no new information to learn that the hero drinks

More ambitious intuition

- Use $K(x)$ to measure expression
- Use $K(x|y)$ to perform filtration
- Use $K(x|y)$ to measure similarity

Objections

- *Objection*: K is uncomputable
- *Objection*: K ignores psychology and aesthetics

Ways to make progress?

- Input into expert testimony in cases involving technical subject matter (e.g., software)?
- Lossy compression, psychology, and aesthetics?

Questions?



$K(x|y)$ = complexity of x not due to y

$K(x) - K(x|y)$ = complexity of x due to y

$K(x|y,z)$ = complexity of x not due to y or z

$K(x|y) - K(x|y,z)$ = complexity of x due to z but not y