HyBR: A Method for Solving Ill-Posed Inverse Problems in Image Processing

JULIANNE CHUNG, JAMES NAGY
Emory University, Atlanta, GA

1 Introduction

Many image processing problems are mathematically modelled as:

\[ b = Ax + e \]

where

- \( b \in \mathbb{R}^n \) represents the observed image
- \( x \in \mathbb{R}^m \) represents the true image
- \( A \in \mathbb{R}^{m \times n} \) models the blurring process
- \( e \in \mathbb{R}^n \) represents noise in the data

What is an inverse problem?
The opposite of a forward problem!
That is, given \( b \) and \( A \), compute \( x \).

What is an ill-posed inverse problem?

![True image](image1)
Blurred & noisy image: \( b \)
Inverse Solution: \( x \)

Remark: Matrix \( A \) is ill-conditioned.
Singular values decay to and cluster at 0, causing amplification of noise in the solution.

2 What is Regularization?

<table>
<thead>
<tr>
<th>Goal</th>
<th>Modify the inversion process to avoid noise amplification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Two examples:</td>
<td></td>
</tr>
</tbody>
</table>
| - Tikhonov Regularization | \[
\min_{x} \| b - Ax \|_2^2 + \frac{\lambda}{2} \| x \|_2^2
\]
| - Conjugate Gradient (CG) | \[
\min_{x} \| b - Ax \|_2^2
\]

Typical Behavior of CG for Ill-Posed Problems

```
Iteration 0
Iteration 10
Iteration 28
Iteration 85
Iteration 150
```

Solution gets better
Noise corrupts!

![Graph](graph1)

```
Iteration 0
Iteration 150
```

Goal: Modify the inversion process to avoid noise amplification

Two examples:
- Tikhonov Regularization
- Conjugate Gradient (CG)

Typical Behavior of CG for Ill-Posed Problems

```
Iteration 0
Iteration 10
Iteration 28
Iteration 85
Iteration 150
```

Solution gets better
Noise corrupts!

4 Conclusions

- HyBR stabilizes noise
- Stopping criteria not as crucial
- Automated HyBR computes fairly good results for any ill-posed problem, with little user input
- Future direction includes incorporating non-negativity constraints

References